AUTOMATED DOCUMENT CASHING SYSTEM

CROSS-REFERENCE TO RELATED APPLICATION

This is a continuation-in-part of copending U.S. application no. 08/866,139, filed May 30, 1997.

5   BACKGROUND OF THE INVENTION

The invention relates to automated banking systems and machines including those which employ or are an improvement over automatic teller machines (ATMs). The invention also relates to providing such

10   ATMs with sufficient security confidence levels with respect to the user, to the document, and to the bank parameters and rules that cash can be securely dispensed to the user as a result of the cashing of payroll or third party remittances or the paying of

15   bills. The confidence levels should be such as would normally be achieved or approach those in comparable transactions with a teller.

A number of security problems arise with the addition to ATMs of functions performed by full service

20   banks and currency exchanges. Such functions include cashing checks and money orders, paying bills, or handling a cash equivalent transaction, such as making a deposit into a bank account. When the bank is to cover such checks and dispense cash to the user, the

25   bank requires validation of the user identity, validation of the genuineness of the document, validation of the amount(s) set forth on the document, validation of a signature on the document, validation of an endorsement when needed, validation of the bank

30   parameters or rules, etc. To date, ATMs have been unable to provide such validations with a reliability

sufficient to cash many documents without the presence of a teller.

To provide an acceptable confidence level to the bank with respect to user validation prior to
5 dispensing cash, a minimum requirement is the use of an ATM card, smart card, or the like, and a password such as a PIN number. The machine could read these, as in conventional machines. In accordance with the preferred embodiments of the present invention, a
10 biometric check also is provided to assure that the person using the machine is a qualified user. This involves extracting recognition features from the user and preferably biometric features such as voice characteristics or features, facial recognition
15 features, retinal features; fingerprint features, palm features; and/or signature features or the like. The qualified user will have previously provided such features to the bank system where they are stored for comparison to the extracted features of the person
20 using the machine. The results comparison must reach certain confidence levels that can be set and/or adjusted by the bank to its satisfaction. Thus, if provided with confidence threshold levels as to card, password and/or the biometric features, the bank can be
25 reasonably assured that the AIM user is a qualified user.

With respect to document validation including the amount of document such as the pay amount of the remittance, a number of validation techniques are
30 desired. To assure that the document being cashed is the original and not merely a photocopy of a valid check having a MICR line thereon, the MICR line should be tested to ascertain that a sufficient magnetic field is present at the MICR line position. Another
35 validation that is desired is a reading of the MICR line contents and communicating to the banking system

that a bank number and an account number for the
identified bank refer to a real rather than a
fictitious bank or account. Additionally, for checks,
it is desired to be able to read the CAR amount and the
5   LAR amount and to compare the same to detect whether or
not the CAR line has been changed, for example, a "1"
has been changed to a "4" or a "7" by merely adding pen
strokes to the "1". Other validations can be used and
obtained to guard against violation of bank parameters
10   or rules.

      Another significant document validation
procedure with respect to checks is a determination
that a signature is present. That is, the check is
signed at the signature line. Going even further, it
15   would be helpful to establish some acceptable signature
confidence level by comparison of the signature against
a stored signature of the user in instances where the
user is signing a check or endorsing the back of the
check. Also, in transactions where the check needs to
20   be endorsed, there should be a validation by the
machine that a signature is present at the endorsement
line. Also, there may be a step of comparing a
signature against a stored signature of the endorser.

      When improper payments are made to the user
25   if the transactional is fraudulent, it is an important
security feature to be able to prove that the user had
an intent to defraud the bank. Absent such proof of
fraudulent intent, the user may escape civil or
criminal liability by claiming that such improperly
30   dispensed cash or cash equivalent was a solely due to
the fault of the ATM or banking system and not
attributable to the user. That is, the user may claim
he did not intentionally cause the cash dispensed or
dispensed in an amount to be larger than that to which
35   he was entitled and that there was no culpability on
his part for the amount of cash dispensed to him.

The wide variety of checks, money orders and bills presents a still further problem with transactions involving cashing of checks or the like, depositing funds to an account, or paying bills. As to

5 each document, the location of the data fields to be analyzed may be different. Preferably, the ATM machine should be able to process large amount payroll checks, smaller amount personal checks, and bills having a bill pay amount located at various places on the bill.

10 Preferably, a cash or cash equivalent dispensing system used without a human teller also is able to meet various bank parameters or rules. Often there is a transaction maximum limit, which may be customized as to the drawer of the check issuer or the

15 payee. The bank may have cash payout limits on a daily or other time basis that should be met with sufficient confidence before dispensing cash. The bank may also have check date rules with respect to processing antedated or post-dated checks that should be

20 satisfied. Finally, the bank may want to set its own thresholds with respect to confidence levels with respect to the identity of the user and validation of document. The system should be able to meet the satisfaction levels desired by the bank, and to be able

25 to adjust such levels for a given transaction, type of transaction, or different validations.

Another consideration for transactions such as cashing checks, paying bills, or other like things from a remote banking machine is the need to make a

30 record and to leave an audit trail for later manual review, if required, of the transaction.

Among some of the mechanical problems that have been experienced with the remote ATM-type machines is that of providing change in coins or small bills.

35 Already, over a single weekend, ATMs are being severely taxed often to the point that they are completely

emptied of their cash contents. In addition, ATMs do
not have change makers. When cashing checks, money
orders or returning change from a cash bill payment,
the ATM must be able to return to the user the exact

5 amount. If the exact amount is in cash, the addition
of a coin change maker and small denomination bill
dispenser adds considerable expense and maintenance
problems to the machine. This would be necessitated to
provide the exact change, including coins, to the user

10 who is cashing a check or performing some other
function, such as paying a bill with cash from which
change is due. The situation is aggravated when the
ATM is performing transactions that include an
automatic fee calculation and deduction of the fee

15 because there will usually be change due for any cash
payout after the transaction fee deduction.

         Another problem with providing a commercially
practical automated banking machine is that of the time
needed for the transactions. Preferably, the

20 transactions should be relatively brief and simple so
that a minimal number of operator actions, such as
touch screen pushes or keystrokes, are required for
each transaction. If a particular transaction takes
more than a minute or two, the system would probably be

25 too slow to adequately service a line of people waiting
to use the machine at a busy time, for instance on a
weekend. Also, if the machine is able to process a
large number of different types of transactions like
those of a full-service bank or a currency exchange,

30 the machine should provide the user a wide range of
funds-delivery or payment options so that the payment
can be made in cash, by credit card, by smart card, or
by withdrawal from a checking or savings account.

         Even if an ATM existed for paying bills or

35 processing checks of various amounts, that ATM might
have difficulty in automatically locating, reading or

interpreting amount lines such as the CAR or LAR, an invoice account number, the amount of the invoice, the amount to be paid, etc. without assistance from the user. Often the numbers written, typed or printed in

5 such lines are relatively small. They might need to be accurately separated from any other writing or numbers to provide a secure and accurate execution of the desired transaction for the document being read. To this end, there is a need for an efficient system or

10 method to locate, read, and interpret such lines with a manual input from the user.

There is a need for an automatic banking machine which includes an ATM-like machine that performs and allows a number of service options, such

15 as for example the withdrawing of cash, the deposit of cash, the cashing of a check, the cashing of a money order, the purchase of a money order, the transfer of funds by wire, payment of a bill and purchase of end user items.

20 SUMMARY OF THE INVENTION

In accordance with the present invention, there is provided an automated banking system including one or more machines which perform the usual ATM functions, but also have such significant security

25 safeguards that they allow the cashing of monetary transaction documents such as checks or money orders, or handling of cash equivalent transactions such as making a deposit in the bank account of the user, without the aid of a teller. These functions are

30 achieved by having sufficient validation of the identity of the user, validation of document, such as being a signed or endorsed check or the like, validation of the amount to be paid in cash or deposited, and validation of the banking system

parameters or rules for the customer and/or transaction.

With respect to validation of the personal identity of the ATM user, a first, minimal fraud

5 protection procedure is to verify that the ATM card and/or the user, as presented at the machine, is associated with a qualified password or PIN number that, upon entry, validates the user as a qualified user. Preferably, and in accordance with the

10 invention, an additional biometric comparison or recognition function is made between extracted features of the user such as face features, voice features, retina features, fingerprint features, palm features, handwriting features for signature verification, etc.

15 In the present invention, the identity of the user is preferably validated with sufficient levels of confidence that cash will be dispensed if the other validation techniques are also satisfied. The bank will have its own rules with respect to how large a

20 transaction will be permitted for the particular user, particularly with respect to the dispensing of cash to the user.

In the preferred embodiment of the invention, the validation of the document preferably includes the

25 extraction of data to compare the LAR amount and the CAR amount. In instances where the check to be negotiated includes a magnetic ink character recognition (MICR) line amount for the amount of the check, the MICR line may be read and a comparison of

30 the LAR to the CAR is not needed.

Additionally, other validation methods for checks may be provided and practiced such as validation that magnetic ink is present on the MICR line and that bank and account numbers are recognized as being valid

35 within the banking system computer system.

To prove that the user intentionally
requested the amount of cash being dispensed, the user
must manually enter amounts using a manual entry device
at the ATM, e.g., the pay amount of the check, so that

5   user will not be able to contend later that a machine
error caused a specific payment to him. A part of the
proof of the intentional request yielded by scanning
the check and presenting a computer-generated image of
the check to the user and prompting the user to enter

10 the payable amount via an entry device.

A still further validation technique is used
in the preferred embodiment of the invention to
safeguard the assets of the bank. Banks may have their
own set of parameters or rules governing payouts and

15 other transactions that must be validated. For
example, validation techniques are used to assure that
the amount of cash being paid out is equal or less than
the transaction or daily limit for the user and the
bank is satisfied with paying out those amounts based

20 on credit history of user.

In accordance with a further aspect of the
invention, the bank will receive a validation that a
signature is present at the signature line of the
document, such as a check, before performing the

25 requested financial transaction with respect to the
check. To this end, the signature line is located and
an analysis is made to an acceptable confidence level
that a signature is present at the signature line. If
a signature is lacking, the check will be rejected.

30 Preferably, an analysis will be made as to verify the
user's signature against stored user signatures to
provide an additional security check to provide further
confidence to the bank doing the transaction. Machine
protection against a skilled forgery is difficult with

35 current technology; nonetheless, unskilled forgeries or
ambiguous signatures may still be detected. In

instances where a third-party check or money order is
to be processed and the ATM user must endorse the
instrument, it is preferred to locate the endorsement
line and at least validate that an endorsement is
5   present in order to protect the receiving bank and
others in the check reconciliation process against
certain types of claims. Again, if the user has
signatures of record, the endorsement can be compared
to the signatures of record and a confidence level
10   validation can be achieved if the transaction is to be
completed.

        In the preferred ATM machine, the user
manually selects the transaction, for instance from a
list of transactions including check cashing, check
15   deposit, bill payment, etc. The user then further
operates the machine by inserting the document into the
machine to cause a computer generated image to be seen
by the user and to allow for analysis of features of
the document image reflective of the document's
20   contents. Because of the wide variety of document
sizes and the variety of locations of the amount line
or lines such as CAR, LAR or bill payment due, it is
preferred to prompt the user to locate the coordinates
of and/or to bound one or more fields for analysis and
25   validation. These fields may include a date field, a
CAR field, a LAR field, an amount field, an account
number or MICR line field. If the document fails to
meet the threshold validity for any one or more of
these bounded fields, further transaction processing is
30   aborted without any cash being dispensed to the user.

        In accordance with a further aspect of the
invention, the ATM user is prompted by the display and
the display provides a bounding box image. The
bounding box can be adjusted by the user who then
35   accepts or rejects with respect to a particular line.
The accepted line in the bounding box is machine

interpreted by OCR or some other image processing technique or the like. Typically, account numbers for bills and the amount of the bill to be paid are located often arbitrarily at various places. They are

5 difficult to locate and must be precisely delineated from other adjacent typing, printing, letter or cursive to allow the transaction to be accomplished. In a preferred embodiment of the invention, the user is prompted to touch a touch screen display at the desired

10 location, e.g., the account number on an invoice. The user then has the option of "tweaking" or adjusting the bounding box to cover only the desired information.

The user is prompted to point to the general area of the document image that contains the

15 information, such as an account number or an amount, to be bounded. The identified region would have its image zoomed on the screen. The first zoom step might be 1.8x linear magnification with the next step 1.1x. The magnification factor would decrease for each additional

20 step to help avoid zoom overshoot. When zooming has been completed, the user would so indicate to the machine and then would be prompted to define the bounding box. This would be done in part by pointing to the beginning and the end of the area of interest.

25 After this first bounding box is generated, a pixel analysis routine would be executed in the pixels at the bounding box borders. This would help ensure that no stray or extraneous characters were inadvertently included in the bounding box leading possibly to a

30 spurious result from later analysis of the data contents of the bounding box. Finally, the user would indicate her acceptance or rejection of the final bounding box, which might change color for clarity, by appropriate keystrokes or touch screen entries or the

35 like.

This technique would avoid problems of lack of bounding box resolution due to a user's finger obscuring a feature of interest during box definition. An alternative bounding box technique would require the

5   user to trace her finger around the region of interest thereby enclosing it rather than simply identifying the beginning and the end of the field.

In order to assist the user the ATM provides prompts to the user and has buttons or touch screen

10   areas that allow the user to switch back to the menu screen to begin again. In the alternative they would allow the user to undo the current screen and go back one screen to make a revision or the like where appropriate.

15   When processing a monetary transaction document for routine bill paying, it is preferred to provide a validation of the bill, the user and the monetary transaction document being used to pay the bill or a portion thereof. With a bill-paying

20   transaction, an operative assumption may be made that where no cash or cash equivalent is being paid out to the user, that user lacks an incentive to misrepresent the pay amount on the checks or the like. In the paying of bills, the user will select the bill payment

25   transaction from a list of transactions. The user will be prompted to make one or more manual entries into machine, like the amount of the bill, the amount being paid by the user which should be equal to or less than the check; and the user's account number on the bill.

30   The machine will scan and interpret the user's account number on the bill, the full amount due, and the date field. If the amount being paid is other than the full amount of the bill, a prompt to enter the tendered amount is provided to user on a screen or the like.

35   When the amount of the check or the like from which the funds are derived is greater than the amount being

paid, the user may be prompted to have the remainder of
the funds paid in cash or loaded into a balance of a
debit or a smart card.

      When paying a bill or making a deposit, the
5   amount field of the document is analyzed on the bill or
the deposit slip and compared to the amount manually
entered by the machine user. This provides one
validation procedure. In some instances, when cashing
or depositing a document such as a check, the drawer of
10   the check may have indicated the amount of the check at
a MICR line. For example, large employers may issue
authorized payroll checks for its enrolled employees.
Those payroll checks are issued with a MICR line having
the amount of the check thereon. In such instances,
15   the MICR amount line may be read and used to validate
the document and the amount to be paid without any
comparison of CAR and LAR lines, as is the case for
checks that lack a MICR amount thereon.

      In accordance with an important aspect of the
20   invention, the check, money order or the like is
scanned and an image therefrom is dissected with
extracted image information being obtained for several
recognition fields. The recognition fields are
processed to provide a list of amount results ranked by
25   confidence values. The user-entered amount and these
confidence values are provided to a processor for
transaction arbitration involving cross-validation
according to rules. If there is validation of the
arbitration using the rules, the transaction is then
30   taken, such as cashing a check, paying a bill, or
making a deposit. The usual recognition fields for a
check are the LAR and CAR. When a remittance document
is also provided to the ATM machine for paying a bill
or the like, the remittance document is scanned and its
35   image dissected with one recognition making a deposit
field being the amount for the remittance. A list of

amount results are ranked by confidence levels and they
are provided to the processor for transaction
arbitration under the rules. The remittance amount is
cross-validated with the check amount results in the
5   transaction arbitration; and, upon validation, the
remittance transaction action then proceeds to
completion.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a front view of an apparatus
10   embodying the invention including a left section, a
central section, and a right section;
FIG. 2 is a top plan view of the three
sections of the machine shown in FIG. 1;
FIGS. 2A and 2B are views of an imaging
15   station for scanning a document;
FIG. 3 is a left side view of one section of
the apparatus shown in FIG. 1;
FIG. 4 is a right side view of the central
section of the apparatus of FIG. 1;
20   FIG. 5 is a right side view of the right
section shown in FIG. 1;
FIG. 6 is a enlarged view of the front of the
apparatus of FIG. 1 showing the various insertion slots
or receiving slots on the apparatus of FIG. 1 with
25   identifying indicia thereon;
FIG. 7 is a rear view of the machine shown in
FIG. 1;
FIG. 8 is a flow chart for showing the
operations occurring after insertion of the card and
30   for verification;
FIG. 8A shows the screen with the instruction
to PLEASE INSERT YOUR CARD;
FIG. 8B shows a screen prompting entry of a
user's password;

FIG. 8C shows the progression of the password verification operation;

FIG. 8D shows the screen when an incorrect password has been entered;

5   FIG. 8E shows that the password is not correct and that the card is being retained;

FIG. 8F shows a screen display prompting the user to make a touch screen selection of the language in which the transactions are to be processed;

10   FIG. 9 shows on the screen the money exchange or transactions options available for the user;

FIG. 9A is a flow chart which shows the initial welcoming and the various options available to the user;

15   FIG. 10 is a screen prompting a checking or savings step as part of a transaction;

FIG. 11 is a screen showing different amounts for withdrawal from checking;

FIG. 11A is a flow chart showing the

20   operations for a withdrawal transaction;

FIG. 12 is a view showing the screen of having an amount prompt for a withdrawing from saving transaction;

FIG. 13 is a flow chart with respect to

25   making a deposit;

FIG. 13A is a screen showing the prompt for the source of a deposit into checking;

FIG. 13B shows a screen providing for entry of the amount of a check to be deposited;

30   FIG. 13C is a screen showing a prompt to endorse or sign the back of the check;

FIG. 13D shows the screen with a message of showing progress in confirming;

FIG. 13E shows a screen prompting the user to

35   take a transaction receipt;

FIG. 13F is a screen with respect to a transaction for a deposit into saving;

FIG. 13G is a screen requesting the amount of cash to be deposited;

FIG. 13H is a flow chart showing machine operations with respect to a cash deposit;

FIG. 13I is a screen showing the amount of cash deposited;

FIG. 13J shows a request to deposit the cash into the cash acceptor slot;

FIG. 13K shows a machine verification of completion of the cash deposit;

FIG. 14 is a flow chart with respect to the document scanning and verification operations;

FIG. 15A is a screen that shows an inquiry to the user requesting a decision as to making a further transaction;

FIG. 15B is a screen display of a touch screen version of the screen display shown in FIG. 15A;

FIG. 16 is a view of the cashing check screen;

FIG. 16A is a flow chart showing the operations with respect to cashing a check;

FIG. 16B shows a screen for requesting the manual entry of the amount of the check to be cashed;

FIG. 16C requests the signing of the back of the check;

FIG. 16D is a screen showing a request to re-insert the inverted check;

FIG. 16DD is a screen requesting the user to re-enter the check amount;

FIG. 16E shows a bar graph of the progress with respect to the reading of the check;

FIG. 16F shows a check cashing and the amount that is available to be received in cash;

FIG. 16G shows the completion of the check cashing and the receipt for the amount deposited to the user's account;

FIG. 16H is a touch screen display version of
5   the screen shown in FIG. 16B;

FIG. 17 is a flow chart showing the operations with respect to cashing a money order;

FIG. 17A is a screen shown to the user when cashing a money order;

10   FIG. 17B requests the signing of the back of the money order;

FIG. 17C states that the money order cannot be cashed;

FIG. 18 shows the screen used when typing in
15   the name of the payee with respect to a money order being purchased;

FIG. 18A shows the amount of the money order being purchased;

FIG. 18B is a flow chart showing the various
20   operations being performed when buying a money order;

FIGS. 18C and 18D show the method of payment and the total transaction at the screen that the money order is being printed and the request to the user to take her receipt;

25   FIG. 19 is a screen display for wiring money;

FIG. 19A shows the account to which the money is to be wired and the name of the bank having the account;

FIG. 19B shows and requests the entry of the
30   Federal routing code;

FIG. 19C shows the account number being added;

FIG. 19D shows a screen requesting the amount and shows a service charge;

35   FIG. 19E is a flow chart showing the operations for a wire transfer;

FIG. 19F shows the total of the transaction and requests a selection of the method of payment;

FIG. 20 is a screen showing a number of bills that can be paid through the apparatus;

5   FIG. 20A shows a telephone bill, service charge and total amount to be charged for payment of the telephone bill;

FIG. 20B shows a screen requesting entry of the telephone bill into the scanner slot;

10   FIG. 20C shows the selection of a gas bill for payment as well as a telephone bill;

FIG. 20D requests insertion of the gas bill into the scanner slot;

FIG. 20E shows the payment for a credit card

15 bill;

FIG. 20F shows the amount of payment with respect to the telephone, gas and credit card bills; and the request for the method of payment;

FIG. 20G shows the screen when the bill is to

20 paid by credit card;

FIG. 20H is a flow chart showing the operations that occur during a bill payment;

FIG. 20I shows a screen confirming payment of the bills;

25   FIG. 20J is a touch screen display version of the screen shown in FIG. 20;

FIG. 21 shows a screen for purchase of items such as stamps, smart cards or telephone cards;

FIG. 21A is a flow chart showing the various

30 operations that occur during the purchasing transaction;

FIG. 21B shows a screen displaying request for a purchase of three smart cards and one telephone card;

35   FIG. 21C shows the total transaction and requests a selection of the method of payment;

FIG. 21D shows a screen showing a $25.00 transaction and showing how much has been inserted to pay for the transaction;

FIG. 21E shows that $20.00 has been paid;

5 FIG. 21F shows that $21.00 has been paid;

FIG. 21G shows that $24.00 has been paid;

FIG. 21H shows that the total of $25.00 has been paid and shows a message on the screen to take the merchandise;

10 FIG. 21I is a touch screen display version of the screen shown in FIG. 21;

FIG. 22 is a flow chart showing the various operations with respect to cash payment;

FIG. 23 shows the payment of change either by 15 credit to a card or by a deposit into a bank account;

FIG. 24 is a block diagram of the apparatus shown in FIG. 1;

FIG. 25 is a flow chart of a signature verification and character recognition process;

20 FIG. 26 is a flow diagram showing details of the overall operation of the processor for generalized document handling;

FIG. 27 is a flow diagram showing generalized flows for various types of document processing 25 involving checks;

FIG. 28 is a generalized flow diagram showing steps related to processor operation related to remittance processing or bill payment;

FIG. 29 is a generalized flow diagram for 30 depositing a check;

FIG. 30 is a generalized flow diagram for the processor when cashing a payroll check;

FIG. 31 is a generalized flow diagram for character amount recognition (CAR) and legal amount 35 recognition (LAR) for an instrument being processed; and

FIG. 32 provides details of processor operation for image character recognition of the type including CAR and LAR;

FIG. 33 is a flow diagram for generating a
5 bounding box;

FIG. 34 is a flow diagram for zooming an image in the bounding box;

FIG. 35 is a flow diagram for the bounding box; and

10 FIG. 36 is a generalized flow diagram for check validation using rules.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

As shown in the drawings for purposes of illustration, the invention is embodied in an automated
15 banking system that includes an apparatus 10 having a housing 12 for housing the components of the apparatus 10 which are to receive an ATM card which can be inserted through an insert, slot or opening 14 in a front wall 16 of the housing 12. The insert slot 14
20 will accept the usual ATM card, credit cards, IC cards or smart cards. The card slot 14 is located immediately above an alphanumeric user keyboard 18 and below a user display 20 comprising a touch screen of the type sold by Dyna-Pro under its Model No. DTFP
25 95633. The user keyboard 18 supplies command signals to a microcomputer 21, in this embodiment a 133 MHz Pentium-based personal computer having a 2.1 gigabyte hard disk drive for storing software, a 32 megabyte random access memory for storing instructions and
30 operands, a 133 MHz Pentium microprocessor, an ISA bus, a PCI bus, a serial interface, and a parallel interface. (FIG. 3). The microcomputer 21 executes application software under Windows 95, which among other things, responds to keystrokes on the user

keyboard 18, and signals from other input devices as set forth below. The microcomputer 21 drives the output display 20 in response to the software it is executing and the various signals it receives from the

5 input devices connected to transfer signals to it.

Located immediately behind the insert card slot 14 is a magnetic card reader 22 (FIG. 4) which will read the ATM card, send signals to the microcomputer 21 through a serial communication card

10 21a, and immediately cause initialization, via the microcomputer 21, of all hardware and software parameters for an operation. The touch screen 20 is provided to assist the user in identifying for the machine the area of the image occupied by the account

15 number and dollar amount of a bill, as will be explained. The illustrated keyboard 18 is a very tough, vandal-resistant, alphanumeric industrial keyboard, such as the Model 300 manufactured by Everswitch USA of Silver Springs, Maryland. The

20 preferred display 20 is a flat LCD display panel sold by Sony Corporation. The keyboard and display panels are selected because they are considered to be tough, strong, easy-to-use, and difficult for thieves or criminals to vandalize or to misuse to illicitly obtain

25 funds from the machine. A backup storage device 23 connected to the computer 21 provides further security for the software and data stored on the hard drive.

As shown in connection with the flow chart of FIG. 8 entitled "insert card and verify screen", the

30 user will see on the screen display 20 the welcome message and a prompt to insert the banking (or ATM) card and to verify a user password with the banking network. The user will be prompted to select English or Spanish as the language for the transactions as

35 shown in FIG 8F. The user will then touch the screen

display to select English or Spanish for the
transaction language.

In a card insert routine 300 a test is made
in a step 302 to determine whether the magnetic-striped
5   identification card has been placed in the card reader
22. If it has not, control is transferred to a step
304 prompting the user to insert the card through the
card slot 14. The card is then read in a step 306 and
the user is prompted and enters a password in a step
10   308. A test is made in a step 310 to determine whether
the password is verifiable by the banking network when
communicated over a modem 29. If the password is not,
a test is made in a step 312 allowing the password to
be entered three more times. Assuming three
15   unsuccessful tries in a step 314, an incorrect password
message is displayed and process loops back to the step
308. If the password is found to be correct after step
310 the transaction is proceeded with in a step 316.
If as a result of step 308 the transaction is
20   cancelled, control is transferred to a step 320 testing
for whether another transaction has been requested.
This may be done by screen prompts to be answered by
the user as exemplified by the screen displays shown in
FIGS. 15A and 15B. The selection may be made by
25   keypads 26 and 27, as shown in FIG. 15A or by touch
screen contact with the appropriately labelled portion
of the screen display shown in FIG. 15B. If it is, a
service option screen 322 is displayed. If it is not,
a test is made in a step 324 to determine whether the
30   card is in the card reader 22. If the card reader 22
does not have a card in it the welcome screen is
displayed in a step 326. If the card is in the card
reader 22 it is ejected back to the customer in a step
328. In the event that the password is entered more
35   than three times control is transferred to a step 330
causing the card to be eaten or retained and placed in

a card bin.   In a step 332 the message is displayed on
the touch screen that the card has been retained and
the touch screen after that displays the welcome screen
in the step 326.

5   The display shown in FIG. 8A prompts the user
to insert the card.   After the insertion of the card,
the display will prompt the user to please enter the
PIN or password number, as shown in FIG. 8B.   The
processing of the entered password is shown in FIG. 8C.

10   If an incorrect password has been used with the card,
then the screen display will display, as shown in FIG.
8D, the phrase "incorrect password", and prompt the
user to "please try again".   If the subsequent or
second password is incorrect, the machine retains the

15   card and the screen display will show on its face, as
shown in FIG. 8E, the statement that there still is an
incorrect password, and that the card is being
retained.   The card has been "eaten" by the machine.
The card can be retrieved only by contacting the

20   financial institution owning the machine.   Having
verified the card and having verified the password or
PIN number with the banking network over the modem 29
or the like, the machine 10 is ready to proceed with a
transaction.   The modem 29 communicates with the

25   computer 21 through the serial interface 21a to which
it is connected.

          The user display screen 20 will then display
the transaction options available to the user, such as
those shown in FIG. 9 which include 1) withdraw; 2)

30   deposit; 3) cash check; 4) cash money order; 5) buy
money order; 6) wire transfer; 7) bill payments; 8)
purchase (lottery tickets, stamps and telephone cards).
The display shown in FIG. 9 will be on the panel
display 20 and adjacent the pair of flanking additional

35   keypads 26 and 27 (FIGS. 1 and 6), which have arrow
keys which are aligned with these options 1-8.   That

is, the pressing of the arrow key 26a opposite the
number "1)" "WITHDRAW" on the screen 20 will initiate a
withdrawal. Whereas, the operation of the second arrow
key 27a _(27b ?)_ in the right hand bank of keys will
5   initiate a "BUY MONEY ORDER" operation, to be described
hereinafter.

Assuming the user has selected the "1)"
withdrawal option by depressing the arrow key 26a
opposite number "1) WITHDRAW", the screen display 20
10   will then display a request to an account for a
withdrawal, i.e., from a checking or savings account.
This is shown in FIG. 10 with the display of a "1)
CHECKING" and a "2) SAVINGS" on the screen display
opposite the arrow keys 26a and 26b. Assuming that the
15   user wishes to withdraw money from a checking account,
the user will press the arrow key 26a. The screen
display 20 will then show the display of FIG. 11 with
the display labeled "WITHDRAW FROM CHECKING" and with
the monetary amounts "20", "40", "50", "100", "200" and
20   other listed opposite the selection arrow keys 26a-26c
and 27a-27c, respectively. By operating one of the
particular arrow keys 26 and 27, i.e., the arrow key
$20.00 for withdrawal from checking, will signal other
positions of the apparatus 10 to perform a number of
25   operations shown on the flow chart entitled "WITHDRAW
screen" shown in FIG. 11A.

In a step 340 the withdraw screen is engaged
and in a step 342 the user is prompted by the screen to
insert the card and a verify screen is displayed. If
30   the card is verified control is transferred to a step
344 allowing the user to choose from a present
withdrawal amount. If the user chooses to cancel the
transaction control is transferred to a step 346
testing for another transaction. If the user chooses
35   not to choose from a preset withdrawal amount, the user
may enter the withdrawal amount in $5.00 increments in

a step 348 or may cancel the transaction and proceed to the other transaction test step 346. Assuming that the withdrawal amount has been entered in $5.00 increments, the withdrawal transaction is performed in a step 350

5   by checking over the banking network. In a step 352 a cash dispenser 30 dispenses the withdrawn amount and in a step 354 the receipt is printed by the receipt printed. Control is then transferred to the step 346 testing for additional transaction prompts. If there

10 is, the service option screen is then displayed in a step 360. If not, the card is ejected from the card reader 22 in a step 362 and the welcome screen is displayed in a step 364.

A connection will then be made by the

15 electronics network and modem 29 via the banking network to access the customer's account in the bank; and then there will be an operation of the cash dispenser 30 (FIGS. 1 and 5) to dispense $20.00 in cash. The cash dispenser communicates with the

20 computer 21 through the serial communication device 21a to which it is connected, as shown in FIG. 24.

The cash dispenser 30 herein is a typical cash dispenser unit used in an ATM machine. The illustrated cash dispenser is a G & D America, Inc.

25 Model ACD which is made by Giestcke and Debrient America, Inc. The illustrated cash dispenser 30 has four (4) bins. Each bin can hold four hundred notes. The preferred cash dispenser 30 is loaded with four hundred $5.00 notes in one bin. The other three bins

30 are each loaded with four hundred $20.00 notes. Manifestly, more or less bins may be used and also different cash dispensers may be used than that described herein.

The illustrated and preferred cash dispenser

35 30, as shown in FIG. 5, is mounted for sliding horizontally to the right for reloading, and is slid

back into the position shown in FIG. 5 where it is
supported on slide tracks 32 mounted on the housing 12.
The cash being dispensed drops through a chute 36 into
a hopper 38 having a pivoted axis door 40. The pivoted
5 access door 40 allows the dispensed cash to drop into a
dispensed cash bin 42. As shown in FIG. 6, in order to
withdraw dispensed cash the user will reach through a
cash bin window 46 in the front housing wall 16 and
remove the cash from the bin 42. As shown in FIG. 7A,
10 access to the interior of the housing 12 and to the
cash dispenser 30 for the replenishing the cash is
through a rear housing door 44. The rear housing door
44 has a double security lock 47a and 47b and a handle
48. With the rear housing door 44 open, the cash bins
15 can be accessed and slid along the tracks 32. The
double security lock 47a and 47b provides security for
the cash sections in the normal manner of an ATM.

     If the user had chosen the "SAVINGS ACCOUNT"
on the display 20 for withdrawal transaction (shown in
20 FIG. 10), she would have pressed the arrow key 26b
opposite the "SAVINGS ACCOUNT" prompt on the screen
display 20. As shown in FIG. 10, the display 20 would
then show the withdrawal from savings screen having the
prompt "WITHDRAW FROM SAVINGS." The user is requested
25 to enter the amount in $5.00 increments of the amount
to be withdrawn. In this instance, the user operates
the keyboard 18 to type in $500.00, the amount to be
withdrawn from savings. In such event, the withdraw
screen under the control of the microcomputer 21
30 executing the steps of the flow chart shown in FIG. 12
used to perform the withdrawal from savings by the
modem through the banking network, and the cash
dispenser 30 is then operated to dispense the cash into
the cash bin 42 for removal by the user.

35   For either a withdrawal from savings or a
withdrawal from checking, it is preferred to print out

a receipt with a receipt printer 50 shown in FIGS. 1
and 3. The receipt printer 50 is connected to the
computer 21 through a parallel communication device 51.
The receipt printer 50 dispenses a printed paper

5   receipt which is fed therefrom and is issued, in this
instance, from a receipt dispensing slot 52 in the
front wall 16 of the housing 12. The user will then
receive the receipt which shows not only the amount
being withdrawn but also the transaction fee. Thus,

10   the total withdrawn from checking or savings for the
transaction will include not only the cash dispensed
but also the transaction fee, i.e., $1.00 per
transaction.

     The illustrated receipt printer 50 is

15   preferably a Model MP342F, manufactured by Star
Micronics America, Inc. of Piscataway, New Jersey. The
receipt printer 50 has an automatic cutter for cutting
the receipt after printing. Manifestly, other printers
or receipt generators may be used than the model

20   described herein.

     The welcome screen is displayed in a step
220, as shown in FIG. 9A. In a step 222 all hardware
and software parameters are initialized. In a step 224
the service options screen is displayed, allowing a

25   choice to enter. The withdrawal screen 226, the
deposit screen 228, the check cashing screen 230, the
cashing of money order screen 232, buy money order
screen 234, the wire transfer screen 236, the bill
payment screen 238 or a make purchase screen 240.

30   Assuming now that the user had selected the
deposit #2 option as shown in FIG. 9, and wanted to
deposit into the checking or savings account, the user
would have pressed the arrow key 26b of the keypad 26,
which is opposite "DEPOSIT." This action results in a

35   request whether to deposit into a checking account or
into a savings account. Assuming the deposit is to be

made into the checking account, the flow chart of FIG.
13 shows the steps performed by apparatus 10 which will
be described in greater detail hereinafter.

5   The deposit screen, which is displayed in a
step 380, requests insertion of the card and displays a
verify screen in a step 382. If the card is not
inserted control is transferred to a step 384 testing
for whether any other transaction is to be carried out.
If it is, in a step 386 the service option screen is
10   displayed. If not, in a step 388 the card is ejected
and the welcome screen is displayed in a step 390. In
the event that the card has been verified a prompt is
made to the user in the step 392 as to the type of
deposit. If the user elects to cancel the transaction,
15   control is transferred to the step 384. If the user
selects "Cash", a cash deposit screen is displayed in a
step 394. If they select "Checking", a check deposit
screen is displayed in a step 396 and if they choose
"Money Order," a money order deposit screen is
20   displayed in a step 398. Control is then transferred
to a step 400, causing the selected transaction to be
performed by the modem 29 through the banking network.
In a step 402 the receipt is printed out and control is
then transferred to the other transaction test step
25   384.

The deposit into checking screen display
(FIG. 13A) prompts the user with the statement: "WHAT
WOULD YOU LIKE TO DEPOSIT IN YOUR CHECKING ACCOUNT 1)
cash; 2) check; or 3) money order". Assuming that the
30   user has elected to deposit a check, the check
transaction will be selected by pressing the arrow key
26b of the keypad 26. As shown in FIG. 13B, a request
then will appear on the screen display 20 labeled "
DEPOSIT CHECK" opposite a window 52 for the amount of
35   the check. In the window 52, the operator will then
use the keyboard 18 to enter the deposit amount of

$675.52. In this instance, a service charge in the amount of $1.00 will also be displayed, as shown in FIG. 13B to the user. If the user has not endorsed the check, the user will see, upon entering the amount,

5   will be that shown in FIG. 13C, which will request the user to "sign the back of the check", and "when ready to insert the check into a scanner slot".

A scanner slot 54 is located above the user display 20, as shown in FIGS. 1 and 6. In this

10 instance, the check will be inserted vertically. The illustrated scanner slot 54 is approximately 4" x 9", and the inserted check will be scanned while it is in this vertical position, as will be described hereinafter. As the check enters the scanner slot 54,

15 it is gripped by feed rollers and moved along a feeding track 56 (FIG. 2). The check feeds directly into and stops at an imaging station 55 where the check is scanned or images of the front and the back sides of the check are captured. A scanning and confirm flow

20 chart is shown at FIG. 14. It will be described in greater detail hereinafter with respect to the software control and operations of the machine. As shown in this flow chart, an optical character recognition (OCR) scanner scans the document. A magnetic ink (MICR)

25 reader reads the magnetic ink data on the check, which will include the bank's identification number as well as the user's checking account number with the bank.

Also, while the check is in this stopped position, its legal line (LAR) will be scanned, and the

30 CAR line will be scanned to verify that the check is for the correct amount, in this instance $675.52. Also, while in the vertical stopped position, it is preferred to have a camera unit 58 and 60 (FIG. 2) disposed on opposite sides to capture images of both

35 sides of the check and connected through a SCSI device 59 to the computer 21. The images are stored on a

magnetic recording medium in TIFF format and are
provided with a tag so that the image file, as shown in
FIG. 14, can be later accessed if so desired.

5   At the beginning of the scanning operation,
the check image is processed to ascertain if the check
has been inserted correctly. In the scanning operation
420 the document is inserted in the scanner slot in a
step 422. The scanner using the camera 58 and 60 scans
both sides of the documents and reads the magnetic ink
10   via a magnetic transducer in a step 424. The document
is placed in the holding area in a step 426 and a
determination is made in a step 428 as to whether the
document is a check or money order on the basis of the
presence or absence of the magnetic ink data. A check
15   is also made in a step 430 to determine whether the
document is inserted correctly. If it is not, the
document is ejected from the document slot 54 in a step
432 and the touch screen 20 displays if the document is
inserted incorrectly in a step 434 following which
20   control is transferred back to the step 422.

If the document is not a check or money order
as determined in a step 428, control is transferred to
a step 440 causing both sides of the document to be
saved in a tagged image file format. If the document
25   was inserted correctly as tested for in step 430, both
sides of the document are saved in a step 440. In a
step 442, the images are analyzed by amount recognition
software of the types supplied by Mitek of San Diego,
California, in particular its Quickstrokes Version 2.5
30   software. Control is transferred to that software from
step 442 and as may best be seen in FIG. 25, in a step
450 the software is run. In a step 452, the software
recognition device is created and initialized. The
form files are read in a step 454, which form files
35   include the positions where the courtesy amount
recognition (CAR) and where the signature is likely

stored in the fields within the document. In a step
456 the scanned image file is read and in a step 458
the neural network contained within the Quickstrokes
software recognizes the characters written in the

5   signature line as well as the characters written in the
courtesy amount recognition (CAR) space and in the
amount recognition (LAR) line. The recognized
characters are then evaluated from the standpoint of a
confidence level in a step 460, and character strings

10   representative of those characters are returned to the
software set forth in FIG. 14 for further evaluation.
Referring now to FIG. 14 in a step 470, the strings
representing the signature verification as well as the
amount on the document are forwarded to the bank

15   network by the modem 29 for confirmation for payout.
If there is no confirmation control is transferred to a
step 472 causing the document to be ejected from the
document slot and in a step 424 a document rejection
message is displayed. In a step 476 the current

20   transaction is denied. In the event that the documents
are confirmed in a step 470, the check or money order
is stacked in an accepted documents bin in a step 478
and confirmation of the current transaction is sent to
the banking network in a step 480.

25   If the images are not stored, the check is
carried around the U-shaped feed path 61 back to an
eject slot 61a in the housing wall 14 for retrieval by
the user. The eject slot 61a is parallel with and to
the left of the insert slot 54. Assuming that the

30   check has been re-inserted correctly and images of both
the front and back have been captured, then the check
is sent to an escrow or holding area 64 in the check
feed track. The holding area 64 communicates through
the serial communication device 21a with the computer

35   21, as shown in FIG. 24.

As best seen in FIG. 4 at the escrow area 64,
the check is held for either depositing into a store
bin 66 if the check has been qualified and accepted, or
the check depositing transaction, the check will be fed
5   from the escrow area back to the eject slot 61a for
removal by the user if failure to verify the signature
causes the check to be rejected for deposit. Assuming
that the banking network has been connected by the
modem 29 to other portions of the apparatus 10 and that
10 the check has been verified, the amount deposited is
sent over the banking network to the identified bank
and identified account of the user for deposit. The
receipt printer 50 is then operated to provide a
written receipt to the user showing the amount
15 deposited minus the transaction charge of $1.00.

        Referring now to FIGS. 2A and 2B, the
document handling of a money order or a check will now
be described in greater detail. The check is inserted
vertically through the scanner slot 54 and passes in
20 front of a pair of first infrared sensors 101 and 102,
which sense that the check has been inserted. These
sensors are on opposite sides of a guide or feed track
100 which includes a pair of spaced parallel plates 103
and 103a extending inwardly to the imaging station 55.
25 Immediately beyond the infrared sensors 101 and 102,
which detect the insertion of the document, is a
pressure roller 105 to push the check against the plate
103. The check is pushed forwardly past a set of
infrared sensors 110 and 112, which will detect when
30 the check is fully inserted into the scanner slot and
is gripped by a feeding belt 112 that runs through an
entry slot 114 between the image scanners 58 and 60 at
the imaging station 54. The feeding belt 112 extends
through imaging station to a large diameter roller 121
35 (FIG. 2B). The check pauses in its travel at the
imaging station 54, where the image taking video or

other scanners 58 and 60 take images of the front and
back of the check. Optical character recognition
readers read the magnetic ink recognition characters
for the bank and for the customer's account.

5   Electronic signals from the image takers 58 and 60
provide information concerning the signature for the
check, the legal line and the amount written thereon,
and the CAR line and the amount written thereon, all of
which are stored magnetically, in this instance, and

10   provided with tag number for later recapture.

          As best seen in FIG. 2B, a U-shaped track 120
is provided around the large diameter roller 121 to
guide the check to reverse its direction of travel and
to move it into a slot between plates 122 and 123 of

15   the check guide track 100 to a pair of inlet infrared
sensors 125 and 126, which sense the check coming into
the inlet of the escrow area 64. The feeding belt 112
is a cogged timing belt which carries the checks about
the drum 121 and between the plates 122 and 123 to the

20   inlet to the escrow area. The cogged feeding belt is
driven by a stepper motor and travels about guide
rollers 127.

          At the escrow or holding area 64, there is
provided a large belt driving drum 130 which drives a

25   cogged feeding belt 131 for conveying the check first
upwardly and to the left into the holding area and from
the latter into the deposit bin 66 above the holding
area 64. If the check is to be rejected, the feeding
belt 131 reverses its direction of travel to eject the

30   check through the eject slot 62. The driving roller
130 includes a stepper motor 132, which is mounted on
the top of the roller 130. The stepper motor 132 is
reversible in its rotation for rotating a drum 130 and
the feeding belt 131 in opposite directions and through

35   a controlled distance.

Infrared sensors 125 and 126 sense the
passage of the check from the imaging station 55 into
the escrow area 64. The feeding belt 131 is guided
along and travels past a series of guide rollers 134a,
5   134b, 134c and 134d to the top of the holding area.
The endless timing belt 131 turns about the top guide
roller 134d and travels downwardly and to the right
past a roller 136 to return to a side of the drum 130,
as seen in FIG. 2A.

10   The check is pushed against the timing belt
131 to travel with the timing belt by four sets of
pressure rollers 140a, 140b, 140c and 140d. At the top
of the holding area is another pair of infrared sensors
141 and 142, which sense the arrival of the upper edge
15   of the check and they signal that the check has been
moved completely into the holding area with the lower
end of the check being at or above the rollers 140a and
134a at the bottom of the holding area and aligned with
the eject slot 62. Once the check has been accepted,
20   the stepper motor 132 is turned to drive the drum 130
and the feeding belt 131 to cause the check to travel
upwardly into the overhead deposit bin 66. On the
other hand if the check is rejected as being
unacceptable, the feeding belt 131 travels in the
25   opposite downward direction to push the lower edge of
the check through the eject slot 62 and return it to
the user. A lower end of the guide plate and a spring
guide finger 147 guide the outgoing ejected check to
slide and travel along a short guide plate 148 to the
30   aligned eject slot 62. Infrared sensors 150 and 151
(FIG. 2A) at the bottom of the holding track sense when
the check has been removed from the eject slot by the
machine user.

During the deposit transaction, the screen
35   display 20 will show a confirming message, such as
shown in FIG. 13D, in the form of a bar that progresses

from left to right in window 69 being viewed by the user. As the receipt is generated by receipt printer 50, the screen display 20 (FIG. 13E) will show that $674.52 "WILL BE DEPOSITED INTO YOUR ACCOUNT. PLEASE

5 TAKE THE RECEIPT WITH YOU."

If, rather than depositing the check into a checking account, the user had selected the option to deposit into a savings account, the screen would display the deposit into savings account shown in FIG.

10 13F. Then, the user would press the arrow key 26b for the "CHECK"; and the check would have been deposited in the same manner as described above with respect to a deposit into a checking account. A cash receipt would have been provided to the user, as was the cash receipt

15 generated for the deposit into the checking account.

Assuming that the user had decided to deposit cash into checking and had pushed the #1 cash button 26a of the keypad for the display screen of FIG. 13A or had pressed the same button for a cash deposit into

20 savings (FIG. 13F), the processor would follow the steps of the cash deposit flow chart shown in FIG. 13H.

In the cash deposit process 500 as set forth in FIG. 13H a cash acceptor 62 is initialized in a step 502. Currency is inserted in the cash acceptor 62 in a

25 step 504 and is accepted thereby. The bills are read and are transferred to a deposited cash bin in a step 506 and the total of the bills presented added up in a step 508. If the user elects to deposit more bills in the cash deposit in a step 510 control is transferred

30 back to step 504. If not, control is transferred to a step 512 where the deposit transaction is proceeded with.

The user display 20 as shown in FIG. 13G for deposit cash would display the prompt "PLEASE INSERT

35 YOUR BILLS INTO THE ACCEPTOR SLOT 60, WHICH IS SHOWN IN THE RIGHTHAND SECTION ABOVE THE CASH DISPENSER." As

may best be seen in FIG. 5, the cash dispenser
accepting slot 60 leads into a cash acceptor module 62,
which accepts cash, specifically bills in denominations
of $1.00, $5.00, $10.00 or $20.00. As shown in FIG.

5   24, the cash acceptor module 62 is electronically
connected to the computer 21 via a resistor network 62a
having a plurality of current limiting resistors. The
resistor network 62a is connected to a digital I/O
board 62b, in this embodiment a National Instruments

10 PC-DIO-96. The digital I/O board 62b is coupled to the
computer 21. The cash acceptor module 62 counts the
deposited bills and has a bin in a hopper 64 to receive
the counted bills. The cash acceptor module 62 is
pivotally mounted at 66 to be swung to a dotted line

15 position for emptying deposited bills therefrom. The
preferred cash acceptor module 62 merely stacks the
inserted bills and counts the same. The cash acceptor
module 62 is preferably a Mars Electronics?
International Cash Acceptor Model AL4-L1-U1M, which is

20 one of several available cash acceptors. It will not
only stack the bills and retain them in the machine 10,
but will add up the total amount of cash. The cash
flow chart shown in FIG. 13H will be described in
greater detail hereinafter in connection with the

25 software and overall control of the machine. The
deposit transaction proceeds from the flow chart of
FIG. 13H back to the flow chart of FIG. 13 to proceed
through the modem and banking methods to make the
deposit into the user's checking or savings account.

30 The machine 10 will operate the receipt printer 50 to
print a receipt to be dispensed to the user through the
receipt slot 52, showing the amount deposited less the
transaction fee, which is illustrated as $1.00 in this
instance.

35   When depositing cash, the illustrated cash
acceptor 62 will total the cash received and show this

cash being deposited, as shown on the screen 20 which
shows that the $20.00 has been deposited after $45.00
more dollars have been deposited, making for a total
deposit of $65.00, as shown in FIG. 13J. A receipt
5   will then be printed by the receipt printer 50, and the
user will be notified that $65.00 will be deposited in
the user's account (FIG. 13K).

     Assuming that the user, when prompted by the
options screen of FIGS. 3 and 9, has elected to press
10   the arrow key 26c to initiate the check cashing
transaction, the user display 20 will prompt the user
to enter the amount of the check into the window 68
(FIG. 16). The flow chart, with respect to cashing a
check, is shown in FIG. 16A.
15   The cash check process is entered at a point
520 and as a result, the magnetic card reader 22
accepts the magnetic identification card in a step 522
and displays a verify screen. The user can exit the
transaction by transferring to a step 524 where he or
20   she is prompted for another transaction. If not, the
amount of the check is entered in a step 526 and the
check is scanned and confirmed in a step 528 as set
forth previously. The user then enters an amount in a
step 530 to be received in cash and the banking network
25   is accessed in a step 532 to determine whether their is
a balance from which the check may be cashed. If so,
in a step 534 the cash dispenser dispenses cash in the
cash amount and in a step 536 the receipt is printed by
the receipt printer. Control is then transferred to a
30   step 524 and if another transaction is desired, the
service option screen is accessed in a step 526. If
another transaction is not wanted, control is
transferred to a step 528 causing the card to be
ejected from the card reader and in a step 530 the
35   welcome screen is displayed.

The user enters through the keyboard 18 the amount, such as $90.00, shown in FIG. 16B, the amount will be scanned and confirmed, and the service charge of $1.00 is shown on the screen display of FIG. 16.

5 The user may select to continue the transaction or to cancel it by pressing the appropriate button of keypads 26 or 27. The touch screen display shown in FIG. 16H allows the user to make the selection by touching the portions of the display labelled either CONTINUE or

10 CANCEL. If the user has not signed the back of the check, the user will be requested to do so (FIG. 16C). If the check was inserted backwards, as it is viewed by the scanner, the check will be returned through the rejected material outlet slot 62. The user will invert

15 the check and insert it now in the correct vertical position into the insert slot 54. From there the check will be carried into the scanning imaging station where cameras 58 and 60 will capture the images of opposite sides of the check. The processor 21 by executing

20 document verification software will then analyze the signature image and compare it with the profile signature of the user. Likewise, the processor, by using the verification software, will also read the cursive legal amount (LAR) line and the written

25 numerical amount at the CAR line, as will be described hereinafter in connection with the document verification software in greater detail.

After re-insertion of the check, the user will be requested to re-enter the amount of $90.00

30 (FIG. 16D). The check image will again be processed and if the amounts match the keyed-in amount the user display will show an "OK" for the amount (FIG. 16D). During the scanning and the verification operations with communication to the user's account, through the

35 banking modem, the screen will display "OCR" with a movable bar, as shown in FIG. 16E. The next prompt

shown on this screen will be to enter the portion of
the check amount that the user wants to receive in
cash.   The cash is selected in $5.00 increments.   The
machine then informs the user that any remaining amount

5   of the check will be received in cash (FIG. 16F).   With
reference to the specific example given herein as shown
in FIG. 16F, the user's screen display 20 will show
that there has been a $90.00 check scan with a service
charge of $1.00, leaving a balance of $89.00.   The

10   operator will have used the keyboard 18 to enter the
request for $40.00 cash, in $5.00 increments, as shown
in window 70.   As will be explained in greater detail
in connection with check cashing flow chart of FIG.
16A, the cash dispenser 30 will then be operated to

15   dispense $40.00 into the cash bin 56, which the user
will then remove.   As shown in FIG. 16G, the amount of
$40.00 will be deposited in the user's account through
the banking network; and the receipt printer 50 will
print a receipt for the deposit of $40.00.

20   The cashing of the money order is much like
cashing a check.   It will be described hereinafter in
connection with the flow chart shown in FIG. 17, and in
connection with the screen of FIG. 17A.

          The cash money order process is accessed in a

25   step 570.   The magnetic card is prompted to be inserted
in a step 522 and a verify screen is raised.   If the
user decides to exit the transaction, she may so signal
and control is transferred to a step 574, testing for
whether another transaction is desired.   Assuming that

30   the card is verified and that the transaction is to
proceed, the amount of the money order to be paid out
is entered in a step 576.   In a step 578 the money
order is inserted and scanned and confirmed, and in a
step 580, assuming the confirmation occurs, the user

35   enters the amount for the money order to receive in
cash.   In a step 580 a query is generated by the modem

29 to the banking network to determine whether the
amount of the money order is backed by funds. Assuming
that it is, in a step 584 the cash dispenser dispenses
the cash amount and a receipt is printed in a step 586.
5   Control is then transferred to the other transaction
test step. If another transaction is desired the
service option screen is displayed in a step 588. if
not, the card reader is ejected in a step 590 and the
welcome screen is displayed in a step 592.
10   Assuming that the user, when viewing the
options available (FIG. 9), had pressed the arrow 26d
opposite "cash money order" to institute this
transaction, the user is then prompted, as shown in
FIG. 17A, to operate the keyboard 18 to enter the
15 amount of the money order, which, in this instance, is
$750.00. The screen will also show the transaction
service charge of $1.00 and the available amount of
$100.00 in cash.
          The cash money order screen displays $100.00
20 in a window 71 and prompts the operator to enter from
the keyboard 18 the amount of cash that the user would
like to receive in $5.00 increments. In this instance,
the user has entered $100.00 into the window 71. In a
manner similar to that used for the scanning of the
25 check, the cameras 58 and 60 photograph both sides of
the cash money order and locate the indicia showing the
amount of the money order and read the amount indicia.
The magnetic ink indicia identifying the issuer and the
account of the issuer are read; and the signature on
30 the back of the money order is scanned and confirmed.
Then a communications network via a modem is connected
to the issuer's account, indicating that the
authenticity of the money order is being checked. When
the machine 10 receives signals that the money order is
35 authentic, the cash dispenser 30 is then operated to
transfer $100.00 cash into the cash bin 46 _(56?)_ for

removal by the user. If the user had not signed the
back of the money order, he would have been informed to
reinsert the money order, as shown in FIG. 17B. If the
money order could not be processed, it would be

5   returned through the reject slot 62. The user display
20 would state that the money order could not be
processed and that the user should check with her
financial institution, as shown in FIG. 17C.

        Assuming the user had selected, in FIG. 9,

10   the #5 option of buying a money order by pressing the
right hand button 27a on the keypad, then the buy money
order screens and flow chart would have been operative,
as will now be described. The first prompt shown on
the purchase money order display 20 (FIG. 18), requests

15   the name of the person to whom the money order is to be
paid. In this instance, the name is John Doe, as shown
in FIGS. 18 and 18A. Having operated the user keyboard
18 to enter the payee's name, i.e., "John Doe," the
user will next enter the amount of $500.00, as shown in

20   window 72 in FIG. 18A. The service charge of $0.50 is
shown so that the total amount needed for the purchase
of the money is $500.50. As may best be seen in FIG.
18B, it is preferred to provide the purchaser of the
money order with a number of options for payment

25   including by cash, by credit card withdrawal from an
account of the user, and by a smart card. Or the user
may return to the money order, if he so desires. The
flow chart for buying a money order is shown in FIG.
18B.

30   In a buy money order transaction, the process
is entered via step 600 and the money order recipient's
name is entered in a step 602 or if cancellation is
desired, control is transferred to another transaction
test step 604. Assuming that the recipient's name has

35   been entered, the amount of the money order is entered
in a step 606 and in a step 608 a method of payment is

chosen causing prompts to occur via a cash payment
screen 610, a credit card screen 612, a smart card
payment screen 614 or a balance withdrawal screen 616.
The particular transaction for payment is then

5   processed in a step 618 and the money order is printed
out in a step 620. A receipt is printed in a step 622
and the transaction test 604 is then made. If further
transactions are to occur, the service option screen is
displayed in a step 624. If not, a test is done in a

10 step 626 to determine if the card is in the card
reader. If it is, the card is ejected in a step 628
and the welcome screen is displayed in a step 630.

        The buy money order transaction will be
tagged and, through the banking network, a money order

15 printer 76 (FIG. 1) will print the money order. The
money order printer 76 is disposed, in this instance,
side-by-side with the receipt printer 50, as is shown
in FIGS. 1 and 3 and is connected to the computer 21
through the parallel communication device 51, as shown

20 in FIG. 24. The printed money order is dispensed from
a money order dispensing slot 78, which is adjacent to
the receipt printing slot 72 in the front housing wall
16 of the apparatus 10. The illustrated money order
printer may be similar to the receipt printer 50 and is

25 available from Star Micronics America, Inc., Model
MP3342F. It includes an automatic cutter.

        As shown in FIG. 18C, the user screen display
20 will then display that $500.50 has been withdrawn
from the user's account, and that the money order is

30 being printed. Both a money order and a receipt will
be issued from the money order slot 78 and the receipt
slot 52, respectively.

        If the user had selected the wire transfer
option in FIG. 9 and had depressed the arrow key 27a

35 for wire transfer, the screen of FIG. 19 would be
displayed on the user's display 20 prompting the user

to use the keyboard 18 to enter the name of the person
to whom the money is to be wired. Then the screen
display 20 would request the name of the bank, as shown
in FIG. 19A, which will be entered, such as First

5   American. The next request of the user is shown in
FIG. 19B and that is for the Federal routing code or
the routing for the bank for the transfer. The routing
is to be typed in by the user using the keyboard. The
number "7896654" has been typed in as the federal

10   routing code in FIG. 19B. The account number of the
receiver is then requested, as shown in FIG. 19C. The
account number in this instance is shown as "987-87654"
and has been typed in by the user using the keyboard
18.

15   Having entered the information for the wire
transfer to a specific account, the screen display 20
requests the amount to be sent, which in this instance,
as shown in window 78 is $850.00. A service charge of
10%, or $85.00 of the $850.00 amount charged is shown

20   to the user bringing the transaction total to $935.00,
as shown in window 78a. The flow chart for a wire
transfer of money is shown in FIG. 19E.
          The wire transfer process 640 is started with
a step 642 for entering information related to the

25   transfer related to the bank the transfer is to be made
to as well as the account. In a step 644 the amount to
be transferred is entered. In a step 646 the method of
paying for the wire transfer is selected, causing
control to transfer to a cash payment screen 648, to a

30   credit card screen 650, to a smart card payment screen
652 or to a withdrawal screen 654. Following that, in
a step 656 the selected payment transfer occurs and the
wire transfer occurs via the modem 29 over the banking
network. In a step 658 a receipt is printed and in a

35   step 660 a test is made for whether another transaction
is to occur. If it is, a service option screen is

displayed in a step 662. If it is not, a test is made in a step 664 to determine if the card is in the reader. If so, the card is ejected in a step 666 and the welcome screen is displayed in a step 668.

5   A request for the method of payment which can be any of four different payment methods, is shown in FIG. 19F. In this instance, the options of cash, credit card, withdrawal from my account, or smart card may be selected by operating the appropriate keypads 26

10   and 27 positioned alongside the display 20, shown in FIG. 19F. After selecting the appropriate method of payment, the machine is then connected over the banking network (FIG. 19E) to the bank to deposit $850.00 in John Doe's account no. 987-87654. The receipt printer

15   50 will cause a printout of the receipt showing a payment and wire transfer to John Doe of $850.00 and a total transaction fee of $935.00, the latter may be charged by credit card, smart card, or withdrawal from my account, as shown in FIG. 19E. On the other hand,

20   the user could have deposited cash of $935.00 in the cash acceptor slot 60. The machine 10 would then count the cash and hold it in the cash acceptor 34. Having finished the transaction, the credit card (if used for payment) would be ejected, as shown in FIG. 19E.

25   Returning again to the options available as shown in FIG. 9, if the operator had pressed the key 27c on the keypad 27 to select the "bill payments" option, then a bill option screen (FIG. 20) would have been shown on the user display 20. The bills which may

30   be paid are listed on the display 20, viz., telephone, electric, gas, cable, water and credit cards. The operator will use one of the keypad buttons on keypads 26 and 27 to select from the screen of FIG. 4 the particular bill to be paid. In the alternative the

35   bill payment selection may be made by touching the appropriately labelled region of the menu display on

the touch screen display shown in FIG. 20J. It will be
requested on the user display, as shown in FIG. 20A, to
enter the amount for the bill selected, such as $129.67
for the telephone bill. Then, the telephone bill may
5   be inserted into the scanning material insert slot 54
where the images of both sides of the bill will be
captured. The particular bill payments have to be
qualified with the user's account beforehand, and the
particular bill has to be recognized so that the amount
10   of the bill and the field specifying money owed can be
located as well as the identity of the creditor
company--the telephone company, in this instance. The
verifier will read the customer's account number, the
payee's account number, and the amount of the bill.
15   The position of this data on the bill as well as the
script, font, etc. will vary greatly. To aid in
reading the bill, a keypad may be provided for
operation by the user. Having manually identified for
the processor 21 all of the fields on the image of
20   bill, the interpretation of the field image is done in
the same manner as analyzing a check or money order.
The bill is verified, and if OK, the request is then
stated as to the total amount to be paid for the
transaction. The user then will receive the request to
25   enter the amount to pay on the telephone bill, as shown
in FIG. 20A, which in this instance, is $129.67. The
service charge of $0.60 will be also displayed to the
user on the user display 20 along with the total, which
is shown in the window at the bottom of the screen 20.
30   For instance, the total charge of $130.27 (FIG. 20A) to
pay the particular telephone bill.

When paying a telephone bill the screen 20
will then interrogate the user as to whether she wishes
to pay another bill via an inquiry, such as the inquiry
35   shown in FIG. 20C wherein it is desired to pay a gas
bill of $45.22. The sum of $45.22 is entered by the

user using the keyboard 18. As shown in FIG. 20D, the
user is then prompted to load the gas bill into the
scanner slot 54. The gas bill will be read in the same
manner as the telephone bill was read by the cameras 58
5   and 60. The magnetic or the other optical character
recognition information on the bill will be analyzed to
associate the payment of $45.22 to the appropriate
account to the bill paying network. If the user also
decides to pay a gas bill, the user will press
10   "continue". Herein, the user decided to pay a credit
card bill of $96.82 as shown in FIG. 20E for a third
service charge of $0.60, which will bring the of the
total service charges to $1.80. The total amount of
the three bills, the telephone bill, the gas bill and
15   the credit card bill plus the service charge will be
$273.51.

Next, the method of payment is requested
(FIG. 20F); and if the user elects to pay with a credit
card, she will press the keypad button 26b and cause
20   the screen (FIG. 20G) to be shown on the user panel 20,
requesting that the user insert the credit card bill
into the slot 54. The bill payments have been made
over the bills payment network and the bills will have
been collected in the receiver bin. This process is
25   set forth as shown in FIG. 20H.

The bill payment process 720 is entered by
selecting the type of bill such as telephone bill or
electric bill, to be paid in a step 722. The bill is
scanned and verified in a step 724 and the amount to be
30   paid is entered manually in a step 726. A test is made
in a step 728 to determine whether other bills are to
be paid. If so, control is transferred back to step
722. If not, control is transferred to a step 730,
testing for other transactions. A method of payment
35   inquiry is made in a step 732 and in response thereto,
a cash screen is displayed in a step 734, or a credit

card payment screen is displayed in a step 736, or a smart card payment screen is displayed in a step 738, or a withdrawal screen is displayed in a step 740. After selecting the payment method, the funds are then

5   transferred so that the bill is paid via modem connection in a step 742 and a receipt is printed out in a step 744. If another transaction is desired from step 730, the service option screen is displayed in a step 746. Otherwise, a test is made to determine if

10 the card is in the card reader 22 in a step 748. The card is ejected in a step 750 and the welcome screen is displayed in a step 752.

When finished with the bill payment, the screen display 20 shows that $273.51 has been withdrawn

15 from the account in FIG. 20H with a notation that "your bills are paid." As the flow chart for the bill payment shows in FIG. 20H, the receipt is printed by the receipt printer 50 which then ejects the receipt through the slot 52 to the user. The ATM card is then

20 ejected from the card reader 22 back to the user.

If the user had elected in FIG. 9 to buy lottery tickets, stamps or telephone calling cards, the purchase option would be selected by depressing the keypad button 27d to cause the purchase display screen

25 of FIG. 21 to be present on the user display 20, which shows the option of buying stamps at $6.50 a booklet, a smart card at $5.00 a card, or a telephone card at $10.00 a card. Obviously, the number of items to be purchased could be enlarged to include lottery tickets

30 or other end user items, which could be dispensed easily through purchasing goods dispensing slots 84, 85 and 86 shown in FIGS. 1 and 6 below three goods dispenser units comprising a lottery ticket dispenser 87, a stamp dispenser 88, a telephone calling card

35 dispenser 89 and a smart card transaction vendor or handler 89a, all connected to the digital I/O board 62b

via the resistor network 62a for communication with the
computer 21.   The disposed goods receiving slots 84, 85
and 86 are located in the front wall 16 of the housing
12, and the dispensers for the lottery tickets, stamps,
5   telephone cards or smart card are mounted on dispenser
support rails 90, as best seen in FIG. 3.   The
dispenser support rails 90 allow for sliding movement
of the dispensers so that they can be accessed through
a rear service door 94 (FIG. 7).   The rear service door
10   94 has its own security lock 96 for denying
unauthorized access to the interior of the housing 12
and to the goods dispensers 87, 88, 89 and 89a.   A
central door 97 having a security lock 98 can be opened
to access the central portion of the machine 10 having
15   the checks and the bills 66, the cameras 58 and 60,
etc.   While a variety of good dispensers could be used,
the illustrated dispensers are card dispensers which
are made by Asahi Seiko USA, Inc., Model CD1000.
Manifestly, good dispensers may be used other than
20   those card dispensers herein described by way of
example.

As shown in FIG. 21, the user may select one
or more of the various items to be purchased.   A
telephone card may be selected by pushing the key 26c
25   to select one $10.00 card.   By pressing the "continue"
button, the user is then provided with a screen
display, as shown in FIG. 21B for buying smart cards or
stamps.   In the alternative the touch screen display
shown in FIG. 21I can be used to make the selection by
30   touching the appropriately labelled region of the
screen display.   In this instance, a three **(five?)**
telephone calling card at $10.00 a card and three smart
cards at $5.00 per card; have been selected by
operating keypad button 26b to result in a grand total
35   of $25.00 in purchases.   The next screen to be shown on
the display 20 prompts the user to select the method of

payment for the $25.00 purchase. The user will then operate one of the keypads to select by cash, credit, withdrawal from account or smart card as a payment mode, as shown in FIG. 21C.

5   In this instance, the operator has decided to pay with cash and has punched the arrow key 26a on the keypad 26. The screen shown in FIG. 21D will then be provided on the display 20 requesting the insertion of the cash into the cash acceptor slot 60. The cash is

10   then verified as counted, FIG. 21E shows that the user has inserted only $20.00, which has been accepted by the cash acceptor 64 and counted. The screen will then show to the user in FIG. 21F that the payment of $21.00 is insufficient for the total transaction of $25.00.

15   If the user only inserts another $3.00, the transaction screen will show that the payment is still $1.00 short, as shown in FIG. 21G wherein the transaction is $25.00. If another dollar bill is inserted into the machine 10, then the user will see the screen shown in FIG. 21H,

20   which will inform the user to take his merchandise with him. Dispensing of the merchandise occurs as shown in the flow chart of FIG. 21A, and the machine control 21 operates the receipt printer 50 to print a receipt for the user which will be dispensed at the dispensing

25   receipt slot 52.

             In order to make a purchase, the purchase process is entered in a step 770. The item to be purchased, such as smart card balance, telephone calling card, stamps or lottery tickets are selected in

30   a step 772, or if desired, the transaction can be cancelled, causing control to be transferred to another transaction test step 774. When an item is chosen to be purchased such as a lottery ticket, the quantity of the item is prompted for in a step 776 and entered, and

35   a test is made in a step 778 as to whether another

purchase is to be made. If it is, control is
transferred back to step 772.

    If not, in a step 780 the method of payment
is selected, causing a cash payment screen to be
5   displayed in a step 782 or a credit card screen to be
displayed in a step 784, or a smart card payment screen
to be displayed in a step 786 or a withdrawal screen to
be displayed in a step 788, following which the funds
are accepted and the merchandise, such as the lottery
10 ticket, is dispensed, in a step 790. The receipt is
printed in a step 792 and another transaction is tested
for in the step 774. If another transaction is
desired, the service options display screen is
displayed in a step 794. If it is not, a test is made
15 to determine if the card is in the card reader 22 in a
step 796. The card is ejected in a step 798 and the
welcome screen is displayed in a step 800.

    As above described herein, it is preferred
not to have any coins or coin changers in the machine;
20 and to provide $5.00 bills as the lowest denomination
bills that will be paid out in change. Usually, the
cash payment process will follow the flow chart shown
in FIG. 22.

    In order to effect a cash payment for one of
25 the transactions, such as the purchase of lottery
tickets, transfer of a balance into the smart card or
into a checking account or the like, the process is
entered in a step 810 and the cash acceptor is
initialized in a step 812. The currency is accepted in
30 a step 814 and is totaled in a step 816. The accepted
bills are stacked in the holding area in a step 818 and
a test is made to determine whether the total covers
the transaction amount in a step 820. If it does not,
more money is accepted in a step 814. If the
35 transaction is covered a determination is made in a
step 822 whether change is due. If change is due, it

is given in $5.00 increments with the remainder credited to the smart card in a step 824 and the transaction proceeds in a step 826.

The $5.00 and $20.00 dollar bills available
5   for change are stacked in the four cash bins. If the payment calculation shows that the cash tendered is sufficient for the transaction and that change is due, the change will be in cash in $5.00 increments by operation of the cash dispenser. Alternatively, any
10 remaining change of less than $5.00 will be credited to a smart card or to a bank account to avoid the necessity of storing and handling small denomination bills and coins. The option will be exercised by the user with respect to change as shown on the screen
15 display (FIG. 23). The user can insert a smart card into the card slot 14, and the smart card writer 89a (FIG. 1) will write the change by increasing the balance on the smart card, and then return the smart card to the user. If the user wants to deposit the
20 change into her account, the user will operate arrow key 26b to cause the deposit transaction to occur over the banking network.

Referring now to FIG. 26, in general, the system architecture as far as the document processing
25 is set forth therein. An invoice image is captured in a step 1000 and a check image may be captured in a step 1002. The images are dissected in a step 1006 and a test is made to determine whether the fields within the image, such as the courtesy amount field or the legal
30 amount field in a check or other image character recognition fields, for instance on bills, are valid, that is, can be interpreted as representing valid amount information or the like. Validation may proceed by selection from a variety of recognition engines,
35 such as a numeric image character recognition engine in a step 1014, an alphabetical image character

recognition engine in a step 1016, a courtesy amount
recognition engine in a step 1018, a numeric optical
character recognition engine in a step 1020, an
alphabetic optical character recognition engine in a
5   step 1022, a legal amount recognition engine in a step
1024, an optical magnetic ink character engine in a
step 1026, or a magnetic ink character engine in a step
1028. The various recognition engines that have thus
been selected pass their results, for instance, in
10   terms of confidence levels, to the field validation
step 1006. The field validation step then inputs
information to a transaction arbitration step 1030.
The transaction arbitration step 1030 may also receive
entered field information from the step 1008 or other
15   user-entered information such as user configuration
information. Such user configuration information might
include an ATM card number, an account number, a PIN
number, or biometric data which is supplied to the
transaction arbitration engine. The information is
20   acted upon in accordance with rules in a rules DLL in a
step 1032. An action, such as payment of a bill or
dispensing of cash, takes place in a step 1034.

Neural-network ICR engines trained from
scratch by exposing the engine to a character training
25   set consisting of thousands of discreet images of
characters that point to their ASCII values. The ICR
engine is then required to recognize a new set of
characters that are not part of the new training set.
Character images that are incorrectly recognized by the
30   engine are assimulated into the original training set
and the engine is retrained on the new set. This
process is repeated until the accuracy of the engine
meets certain predefined standards on arbitrary
collections of real world image data, which standards
35   are based upon comparable performance by professional
data entry personnel.

There are a number of character recognition engines that could be employed by CIRS. The ICR engines that could currently be used by CIRS include FieldScript and CheckScript, v2.2. by Parascript

5 (Colorado Springs, Colorado) for LAR; Quickstrokes v2.4, by Mitek(San Diego, CA); OrboCAR v2.13, by OrboGraph (Israel) for CAR, and Wordscan Plus, 1998 edition, by Caere for OCR of machine print.

Referring now to FIG. 27, in general, the

10 types of transactions that may be performed by the apparatus 10 are set forth therein. All of these transactions relate to check processing. Remittance processing involving automated payment of a bill can occur in step a 1036. The remittance processing is

15 followed by check processing in a step 1042. Arbitration and validation follows check processing in a step 1044. An action such as payment of the bill occurs in a step 1046. Proof of deposit may occur in a step 1038. Following which, check processing occurs in

20 step 1042. The arbitration and validation step 1044, and the action step 1046 are then performed. Likewise, a payroll check may be cashed in a step 1040 involving check processing in step 1042. The arbitration and validation of the check processing information occurs

25 in step 1044. The payroll in action step 1046 next takes place.

Remittance processing details are more specifically shown in FIG. 28 wherein, in a step 1050, a user would be prompted to manually enter a full

30 invoice amount, an amount to be paid and an account number at the apparatus 10. That information would be passed to a transaction arbitration step 1076. In addition, the check would be optically and magnetically scanned in a step 1052 to produce imaging of the front

35 and back of the check as well as magnetic MICR information. The images from the check and the

magnetic information would be dissected in a step 1054.
Date amount recognition takes place in a step 1056.
Legal amount recognition occurs in a step 1060.
Courtesy amount recognition occurs in a step 1062.

5   Each of those last three steps would then pass their
results in terms of a confidence level or an output to
the transaction arbitration step 1076. If an invoice
is to be processed as part of the remittance, the
invoice document is optically scanned in a step 1064

10   and the image is dissected in a step 1066. The image
character recognition amount paid field is interpreted
in a step 1068. The optical character recognition date
field is interpreted in a step 1070. The account
number, as sensed by optical character recognition in a

15   step 1072, has its information passed to the
transaction arbitration step 1076. In addition, the
full invoice amount field is optical character
recognized in step 1074, and that information is passed
to the transaction arbitration step which then acts

20   upon it and pays the bill in step 1078.

        Proof of deposit processing is performed as
shown in FIG. 29. In a step 1080 the user is prompted
to enter manually the amount of a check at the
apparatus 10. The check is image scanned in a step

25   1082. The check image is dissected in a step 1084.
Legal amount recognition takes place in a step 1086.
Courtesy amount recognition takes place in a step 1088.
Date recognition would take place in a step 1090. The
results of steps 1086-1090 are passed to a transaction

30   arbitration step 1092. The transaction arbitration
step 1092 would then act in accordance with rules set
forth in the rules module 1094. Action such as
depositing funds and issuing a proof of deposit occur
in a step 1096.

35   As shown in FIG. 30, a payroll check may be
cashed. In a step 1100, the user is prompted to enter

the check amount into the apparatus 10. The entered
amount is passed to a transaction arbitrator step 1112.
In a step 1102, the check is optically scanned and in
step 1104 an image of the check is dissected. In a

5   step 1106 there is a magnetic recognition of the MICR
line, for instance, to determine the bank number, the
account number, and even in some instances the amount
of the check. There is also optical recognition of the
MICR line in a step 1108 and a date amount recognition

10   in a step 1110.   However, that information is passed
to the transaction arbitration step 1112. Following
step 1112, action, for instance, payment of funds, is
taken in a step 1114.

        As shown in FIG. 31, details are set forth
15   for the legal amount recognition and courtesy amount
recognition arbitration procedures. A user enters the
check amount in a step 1120. There is a cross-
validation in a transaction arbitration step 1134.

        A check is optically scanned in a step 1122
20   and its image is dissected in a step 1124. The portion
of the check image related to the courtesy amount is
extracted in a step 1126 by way of bounding box
recognition techniques set forth below. In a step 1127
the image is processed, including by way of character

25   segmentation in a step 1128, and courtesy amount
recognition values and associated confidence levels in
output in a step 1130. The multiple courtesy amount
recognition values may be output in step 1132 ranked
according to their respective confidence levels. That

30   information is passed to the transaction arbitration
step 1134.

        In a similar fashion, the legal amount is
extracted after image dissection in a step 1136. The
image is processed in a step 1138, including via word

35   segmentation in a step 1140, and the legal amount
recognition conclusion is generated in step 1142.

Multiple legal amount recognition values, together with
their respective confidence levels, are transmitted in
a step 1144 to the transaction arbitration step 1134.
In a step 1150 an associated document, which may be a
5   bill or invoice to be paid, will be optically scanned.
The bill or invoice image will be dissected in a step
1152, and the image of the document amount is indicated
possibly by a bounding box and extracted in step 1154.
The document amount would be processed, including by
10   character segmentation in a step 1158 and optical
character recognition of the document amount in the
step 1160. Document amount results are ranked by
confidence level and transferred in a step 1162 to the
transaction arbitration step 1134.
15   In addition, after the associated document
has been scanned, other amounts might be extracted
required by the transaction and passed to the
arbitration step 1134. The image would be extracted in
step 1163a. The image would be processed in step
20   1163b. Character recognition would occur in step
1163c. Optical character recognition of other fields
would occur in step 1163d. The resulting amounts or
character strings would be ranked by confidence value
in step 1163e. The arbitration step would act in
25   accordance with various rules as to confidence levels
and the like in step 1166 and take action, for
instance, related to payment of an amount in a step
1168.
          As shown in FIG. 32, the image character
30   recognition process occurs at a step 1170 in which the
document is scanned, its image is processed in a step
1172, and relevant fields are located in a step 1174.
Characters are segmented in a step 1176 and characters
are classified in a step 1178. The document is scanned
35   in the step 1170 by digitizing the image in a step
1180. During the image processing step 1172, the

image is registered and deskewed in a step 1184. Extraneous lines are removed in step 1186 and noise is removed in step 1188. In order to determine the field location, whether by bounding boxes or the like, the
5   field is matched in some instances with a predefined image character recognition template in a step 1190. In order to perform the character segmentation step 1176, features are extracted in a step 1194, a character string analysis is made in step 1196 and a
10   best fit to a predefined number of characters is determined in step 1198. In order to perform character classification, context analysis is performed in step 1200, including by the use of dictionaries in step 1202 and look-up tables in a step 1204, which information is
15   then relied upon by validation routines in step 1206.

In order to generate a bounding box a bounding box procedure 1300 causes the processor, through the touch screen, to prompt the user to point to the beginning of a field such as a character amount
20   field, a legal amount field, or some other field in a step 1302. Pointing at the touch screen causes the touch screen to signal the processor as to the X and Y coordinates of the point in a step 1304. In a step 1306 the processor is prompted to point to the end of
25   the particular field. In a step 1308 the system records the X and Y coordinates of the end field point identified. In a step 1310 the X and Y coordinates of the initial point and the end point are used to define a region for an initial bounding box. In the step 1312
30   a pixel analysis routine to be described hereinafter determines whether significant portions of characters, strokes or the like extend outside the preliminary bounding box region. In a step 1314 the bounding box is then drawn on the screen and the user is prompted by
35   the processor with a query as to whether they are satisfied with the bounds of the bounding box in a step

1316. If they are not, control is transferred back to step 1302 if they are, the bounding box is adopted in a step 1318 to define the region of interest to be operated upon by a recognition engine or recognition
5   software which preforms optical character recognition, image character recognition, car recognition or the right.

        The bounding box may also be adjusted by a zooming procedure, as set forth in a zooming procedure
10 1330 shown in FIG. 34, in that procedure a document image is displayed on the touch screen in a step 1332. In a step 1334 the user is prompted to locate the field of interest by touching the screen. In a step 1336 the X and Y coordinates of the point touch are recorded in
15 RAM. In a step 1338 the displayed image zooms in on the area around the point magnifying it. A test is made in a step 1340 to determine whether the zoom level is OK. If the user touches the screen further the process loops back to a step 1334 causing further
20 zooming to take place. For instance the first zoom might magnify 1.8 times, the next zoom by 1.1, and successive zooms by smaller amounts so that there is a quasi-asymptotic approach without significant overshoot. Following completion of the zoom the
25 bounding box procedure 1300 is entered by a step 1342.

        In order to preform the pixel analysis step 1312, as is shown in FIG. 35, the initial bounding box is generated on the basis of the user input as represented by a step 1360. In a step 1362 the area
30 immediately surrounding the bounding box within half a character height is analyzed to determine whether portions of characters, cursive strokes or the like extend outside the bounding box region. If so, the bounding box borders are adjusted in a step 1364 to
35 include the extraneous stroke portions within a somewhat larger bounding box. If there is a

characteristic of characters which would be from a
different field, as detected in a step 1366, the
bounding box borders would be shifted to preclude
inclusion of those characters within the field of
5   interest. A test is made in a step 1368 to determine
if field characters are not fully enclosed by the
bounding box and then the borders would be moved so as
to fully enclose all character in the field of interest
in that step. The new bounding box would then be
10   generated in the step 1370 and control would be
transferred to step 1314 on FIG. 33 to draw the
bounding box on the screen.

Transaction arbitration for any of the above
steps takes place as a result of the rules DLL which
15   was provided and is depended thereon. Although variety
of transaction arbitration sets of rules can be created
for various environments, in one embodiment of the
instant application, as is best shown in FIG. 36, the
courtesy address recognition result in a step 1400 is
20   passed to a rate of confidence algorithm in step 1402
which operates on it. The courtesy amount recognition
threshold value for the confidence from the step 1404
is passed to a step to determine whether the rate of
confidence is greater than threshold value in a step
25   1406. If it is not, the transaction is rejected in
step 1408 and no further action is taken. If it is,
control is passed to a step 1410 where rate of
confidence algorithm is applied which receives the CAR
recognition and CAR threshold as well as the numeric
30   LAR recognition result related to the confidence level
in the step 1412. If the overall confidence value is
greater than the threshold in the step 1414 when taking
into account the legal amount recognition threshold
from step 1416, the transaction is accepted in a step
35   1418. If not, it is rejected in a step 1420.

Description of Payroll Check Cashing Routine

```
CIRS_ACTION CIRS_Rules_Payroll (
        CIRS_ENTERED    *pEntered;   // Fields entered by
                                     // the user.
5       CIRS_CHECK       *pCheck;    // Check image
                                     // recognition.
        CIRS_CONFIG      *pConfig)   // Application
                                     // specific parameters.


    {
10            CIRS_RESULT            MagMICR, OptMICR,
                                     CheckDate;
        int                          i, found;
        int                          Dollars, Cents;


    // Establish the list of candidates which pass
15  threshold, for each field.
        CIRS_FilterByConf (&MagMICR, pCheck->MagMICR,
    pConfig->Check.MagMICR.Thresh);
        CIRS_FilterByConf (&OptMICR, pCheck->OptMICR,
    pConfig->Check.OptMICR.Thresh);
20        CIRS_FilterByConf (&CheckDate , pCheck->Date,
    pConfig->Check.Date.Thresh);


    // Reject the transaction if there isn't at least one
    candidate for each field.
        if (MagMICR.CandidateCount <= 0) return
25  (CIRS_ACTION_BAD_CHECK_IMAGE);
        if (OptMICR.CandidateCount <= 0) return
    (CIRS_ACTION_BAD_CHECK_IMAGE);
        if (CheckDate.CandidateCount <= 0) return
    (CIRS_ACTION_BAD_CHECK_IMAGE


30  // At least one good date candidate must be within the
    last 30 days.
```

```
          found = FALSE;
          for (i = 0; i < CheckDate.CandidateCount; i++) {
              if ((CheckDate.Candidates(i).Value.Date.Date
   < time (NULL) &&
5                    (CheckDate.Candidates(i).Value.Date.Date
   > (time (NULL) - 60*60*24*30))) {
                  found = TRUE;
                  break;
                  }
10        }
          if (! found) return (CIRS_ACTION_BAD_CHECK_DATE);


   // At least one good candidate from the Optical MICR
   result must match the Magnetic MICR.
          Dollars =
15 MagMICR.Candidates(0).Value.Amount.Dollars;
          Cents = MagMICR.Candidates(0).Value.Amount.Cents;
          found = FALSE;
          for (i = 0; i < OptMICR.CandidateCount; i++) {
                  if
20 ((OptMICR.Candidates(i).Value.Amount.Dollars ==
   Dollars) &&
          (OptMICR.Candidates(i).Value.Amount.Cents ==
   Cents)) {

                  found = TRUE;
25                break;
                  }
          }
          if (! found) return (CIRS_ACTION_BAD_CHECK_MICR);


   // Check the MICR amount and account against the
30 database.
          if (! CIRS_ValidateMICR (MagMICR.Candidates(0),
   pEntered->Cash.Dollars))
                  return (CIRS_ACTION_BAD_CHECK_MICR);
```

```
// Transaction is acceptable.
        return (CIRS_ACTION_ACCEPT);
}
```

5   Description of Remittance Processing Routine for Bill Payment

The Rules DLL complies to the following API definition:

```
// Confidence level for a char, word or field result.
typedef U16              CIRS_CONF;              // 0-1000.

// Dollar amount field.
typedef struct {
        U16             Dollars;        // 0-9,999.
        U8              Cents;          // 0-99.
        CIRS_CONF       Conf;           // Amount confidence.
} CIRS_AMOUNT;

// Generic account field.  May contain any application-specific
characters, but would typically be digits.
typedef char             CIRS_ACCOUNT[20];

// CIRS specific date field.
typedef struct {
        time_t          Date;
        CIRS_CONF       Conf;           // Date confidence.
} CIRS_DATE;

// Image location coordinates for a char, word or field result.
Measured in pixels.  Coordinate systems
// interpolation between devices is performed in the Field
Validation Module.
typedef struct {
        U16             x, y, dx, dy;   // Device specific.
} CIRS_SEGMENT;

// A single character returned from a recognition engine.
typedef struct {
```

```
         char                  Value;              // [0-9A-Za-z{punct}].
         CIRS_SEGMENT          Segment;            // Character location.
         CIRS_CONF             Conf;               // Character confidence.
    } CIRS_CHAR;


 5  // A single word returned from a recognition engine.
    typedef struct {
         CIRS_CHAR             *Chars;             // List of characters.
         U8                    CharCount;          // Number of chars in
                                                   // word.
10       CIRS_SEGMENT          Segment;            // Word location.
         CIRS_CONF             Conf;               // Word confidence.
    } CIRS_WORD;


    // A single recognition candidate for a field.
    typedef struct {
15       CIRS_WORD             *Words;             // List of words.
         U8                    WordCount;          // Number of words in
                                                   // field.
         CIRS_SEGMENT          Segment;            // Field location.
         CIRS_CONF             Conf;               // Field confidence.
20       union {                                   // Alternate
                                                   // representations.
         CIRS_AMOUNT           Amount;             // Only one of these
                                                   // will exist,
         CIRS_ACCOUNT          Account;            // determined by the
25                                                 // data type
         CIRS_DATE             Date;               // of the field.
         } Value;
    } CIRS_CANDIDATE;


    // A complete recognition result for a field.
30  typedef struct {
         CIRS_CANDIDATE        *Candidates;        // List of candidates.
         U16                   CandidateCount;     // Number of candidates.
    } CIRS_REC_RESULT;


    // Complete set of fields entered by the user at the system
35  console.
    typedef struct {
         CIRS_AMOUNT           Invoice;            // Entered Invoice
                                                   // Amount.
         CIRS_AMOUNT           Paid;               // Entered Paid Amount.
```

```
                CIRS_ACCOUNT      Account;           // Entered Account.
        } CIRS_ENTERED;


        // Complete set of fields recognized from a check image.
        typedef struct {
5               CIRS_REC_RESULT   CAR;               // Check CAR.
                CIRS_REC_RESULT   LAR;               // Check LAR.
                CIRS_REC_RESULT   Date;              // Check Date.
                CIRS_REC_RESULT   Account;           // Check Account.
        } CIRS_CHECK;


10      // Complete set of fields recognized from an invoice image.
        typedef struct {
                CIRS_REC_RESULT   Amount;            // Invoice Amount.
                CIRS_REC_RESULT   Paid;              // Invoice Paid Amount.
                CIRS_REC_RESULT   Date;              // Invoice Date.
15              CIRS_REC_RESULT   Account;           // Invoice Account.
        } CIRS_INVOICE;


        // All actions which can be returned from the Rules DLL.
        typedef CIRS_ACTION       int;
        typedef enum {
20              CIRS_ACTION_ACCEPT,                  // Good transaction.
                CIRS_ACTION_REJECT,                  // Generic failed
                                                     transaction.

                CIRS_ACTION_BAD_CHECK_IMAGE,         // Check image is poor.
                CIRS_ACTION_BAD_CHECK_CAR,           // CAR can't be
25                                                   validated.

                CIRS_ACTION_BAD_CHECK_LAR,           // LAR can't be
                                                     validated.

                CIRS_ACTION_BAD_CHECK_DATE,          // Check Date can't be
                                                     validated.
30              CIRS_ACTION_BAD_CHECK_ACCT,          // Check Acct can't be
                                                     validated.

                CIRS_ACTION_BAD_INVOICE_IMAGE,       // Invoice image is
                                                     poor.

                CIRS_ACTION_BAD_INVOICE_AMOUNT,      // Invoice Amt can't be
35                                                   validated.

                CIRS_ACTION_BAD_INVOICE_PAID,        // Invoice Paid can't be
                                                     validated.

                CIRS_ACTION_BAD_INVOICE_DATE,        // Invoice Date can't be
                                                     validated.
40              CIRS_ACTION_BAD_INVOICE_ACCT         // Invoice Acct can't be
```

```
                                              validated.
    } CIRS_ACTIONS;

    // Confidence threshold used for validating fields.  Threshold
    values are determined experimentally,
 5  // based on a large set of sample images.  Threshold values are
    customized for each application.
    typedef struct {
            CIRS_CONF          Thresh;              // 0-1000.
    } CIRS_CONF_CONFIG;


10  // Set of thresholds for all fields on a check image.
    typedef struct {
            CIRS_CONF_CONFIG          CAR;
            CIRS_CONF_CONFIG          LAR;
            CIRS_CONF_CONFIG          Date;
15          CIRS_CONF_CONFIG          Account;
    } CIRS_CHECK_CONFIG;


    // Set of thresholds for all fields on an invoice image.
    typedef struct {
            CIRS_CONF_CONFIG          Amount;
20          CIRS_CONF_CONFIG          Paid;
            CIRS_CONF_CONFIG          Date;
            CIRS_CONF_CONFIG          Account;
    } CIRS_INVOICE_CONFIG;


    // Complete set of application specific thresholds.
25  typedef struct {
            CIRS_CHECK_CONFIG          Check;
            CIRS_INVOICE_CONFIG          Invoice;
    } CIRS_CONFIG;


    // Single Rules DLL entry-point.
30  extern CIRS_ACTION CIRS_Rules (
            CIRS_ENTERED          *pEntered;   // Fields entered by the user.
            CIRS_CHECK            *pCheck;     // Check image recognition.
            CIRS_INVOICE          *pInvoice;   // Invoice image recognition.
            CIRS_CONFIG           *pConfig);   // Application specific
35                                                 parameters.
```

To facilitate the implementation of the Rules DLL,
an API is provided which supports a basic set of

CIRS related functions. The definition of the CIRS
Support API follows:

```
void CIRS_FilterByConf (
        CIRS_REC_RESULT    *pOutResult,
        CIRS_REC_RESULT    InResult,
        CIRS_CONF          Thresh);
```

Remittance Processing

```
CIRS_ACTION CIRS_Rules (
        CIRS_ENTERED      *pEntered;   // Fields entered by the user.
        CIRS_CHECK        *pCheck;     // Check image recognition.
        CIRS_INVOICE      *pInvoice;   // Invoice image recognition.
        CIRS_CONFIG       *pConfig)    // Application specific
                                       //   parameters.

{
        CIRS_RESULT               CAR, LAR, CheckDate,
                                  CheckAcct, InvPaid, InvAcct;
        int                       i, found;

// Establish the list of candidates which pass threshold, for each
field.
        CIRS_FilterByConf (&CAR, pCheck->CAR,
pConfig->Check.CAR.Thresh);
        CIRS_FilterByConf (&LAR, pCheck->LAR,
pConfig->Check.LAR.Thresh);
        CIRS_FilterByConf (&CheckDate , pCheck->Date,
pConfig->Check.Date.Thresh);
        CIRS_FilterByConf (&CheckAcct , pCheck->Account,
pConfig->Check.Account.Thresh);
        CIRS_FilterByConf (&InvPaid , pInvoice->Paid,
pConfig->Invoice.Paid.Thresh);
        CIRS_FilterByConf (&InvAcct , pInvoice->Account,
pConfig->Invoice.Account.Thresh);

// Reject the transaction if there isnÆt at least one candidate
for each field.
        if (CAR.CandidateCount <= 0) return
(CIRS_ACTION_BAD_CHECK_IMAGE);
        if (LAR.CandidateCount <= 0) return
(CIRS_ACTION_BAD_CHECK_IMAGE);
```

```
        if (CheckDate.CandidateCount <= 0) return
(CIRS_ACTION_BAD_CHECK_IMAGE
        if (CheckAcct.CandidateCount <= 0) return
(CIRS_ACTION_BAD_CHECK_IMAGE);
5       if (InvPaid.CandidateCount <= 0) return (CIRS_ACTION_BAD_
INVOICE _IMAGE);
        if (InvAcct.CandidateCount <= 0) return (CIRS_ACTION_BAD_
INVOICE _IMAGE);


    // At least one good date candidate must be within the last
10  15-days.
        found = FALSE;
        for (i = 0; i < CheckDate.CandidateCount; i++) {
            if ((CheckDate.Candidates(i).Value.Date.Date < time
(NULL) &&
15              (CheckDate.Candidates(i).Value.Date.Date > (time
(NULL) û 60*60*24*15))) {
                    found = TRUE;
                    break;
            }
20      }
        if (! found) return (CIRS_ACTION_BAD_CHECK_DATE);


    // At least one good candidate from the CAR field must match the
Entered Paid Amount.
        found = FALSE;
25      for (i = 0; i < CAR.CandidateCount; i++) {
            if (CAR.Candidates(i).Value.Amount.Dollars ==
Entered.Dollars) {
                    found = TRUE;
                    break;
30          }
        }
        if (! found) return (CIRS_ACTION_BAD_CHECK_CAR);


    // At least one good candidate from the LAR field must match the
Entered Paid Amount.
35      found = FALSE;
        for (i = 0; i < LAR.CandidateCount; i++) {
            if (LAR.Candidates(i).Value.Amount.Dollars ==
Entered.Dollars) {
                    found = TRUE;
40                  break;
```

```
                    }
            }
            if (! found) return (CIRS_ACTION_BAD_CHECK_LAR);

        // At least one good candidate from the Invoice Paid field must
   5    match the Entered Paid Amount.
            found = FALSE;
            for (i = 0; i < InvPaid.CandidateCount; i++) {
                    if (InvPaid.Candidates(i).Value.Amount.Dollars ==
        Entered.Dollars) {
  10                        found = TRUE;
                            break;
                    }
            }
            if (! found) return (CIRS_ACTION_BAD_INVOICE_PAID);

  15        // Transaction is acceptable.
            return (CIRS_ACTION_ACCEPT);
        }


        // Confidence level for a char, word or field result.

        typedef U16                 CIRS_CONF;              // 0-1000.


  20    // Dollar amount field.
        typedef struct {
            U16                 Dollars;                // 0-9,999.
            U8                  Cents;              // 0-99.54.
        The Rules DLL complies to the following API definition:


  25    // Confidence level for a char, word or field result.
        typedef U16                 CIRS_CONF;              // 0-1000.


        // Dollar amount field.
        typedef struct {
            U16                 Dollars;                // 0-9,999.
  30        U8                  Cents;              // 0-99.
            CIRS_CONF           Conf;                   // Amount
        confidence.
```

```
} CIRS_AMOUNT;

// Generic account field.  May contain any
application-specific characters, but would typically be
digits.
typedef char         CIRS_ACCOUNT[20];

// CIRS specific date field.
typedef struct {
    time_t          Date;
    CIRS_CONF       Conf;                    // Date
confidence.
} CIRS_DATE;

// Image location coordinates for a char, word or field
result.  Measured in pixels.  Coordinate systems
// interpolation between devices is performed in the
Field Validation Module.
typedef struct {
    U16             x, y, dx, dy;   // Device specific.
} CIRS_SEGMENT;

// A singe character returned from a recognition
engine.
typedef struct {
    char            Value;          //
[0-9A-Za-z{punct}].
    CIRS_SEGMENT    Segment;                 // Character
location.
    CIRS_CONF       Conf;                    // Character
confidence.
} CIRS_CHAR;

// A single word returned from a recognition engine.
typedef struct {
```

```
        CIRS_CHAR        *Chars;          // List of
   characters.
        U8               CharCount;          // Number of
   chars in
5                                         word.
        CIRS_SEGMENT     Segment;         // Word
   location.
        CIRS_CONF        Conf;            // Word
   confidence.
10 } CIRS_WORD;


   // A single recognition candidate for a field.
   typedef struct {
        CIRS_WORD        *Words;          // List of words.
        U8               WordCount;          // Number of
15 words in
                                          field.
        CIRS_SEGMENT     Segment;         // Field
   location.
        CIRS_CONF        Conf;            // Field
20 confidence.
        union {                           // Alternate
                                          representations.
        CIRS_AMOUNT          Amount;          // Only
   one of these
25                                        will exist,
        CIRS_ACCOUNT     Account;         // determined
   by the
                                          data type
        CIRS_DATE        Date;            // of the
30 field.
        } Value;
   } CIRS_CANDIDATE;


   // A complete recognition result for a field.
   typedef struct {
```

```
        CIRS_CANDIDATE        *Candidates;    // List of
candidates.
        U16               CandidateCount;    // Number of
candidates.
5   } CIRS_REC_RESULT;


// Complete set of fields entered by the user at the
system console.
typedef struct {
        CIRS_AMOUNT        Invoice;              // Entered
10  Invoice
                                        Amount.
        CIRS_AMOUNT        Paid;                     //
Entered Paid Amount.
        CIRS_AMOUNT        Cash;                     //
15  Entered Check Amount.
        CIRS_ACCOUNT   Account;          // Entered
Account.
        } CIRS_ENTERED;


// Complete set of fields recognized from a check
20  image.
typedef struct {
        CIRS_REC_RESULT        CAR;                  //
Check CAR.
        CIRS_REC_RESULT        LAR;                  //
25  Check LAR.
        CIRS_REC_RESULT        Date;                 //
Check Date.
        CIRS_REC_RESULT        Account;              //
Check Account.
30        CIRS_REC_RESULT        MagMICR;            //
Check Amount.
        CIRS_REC_RESULT        OptMICR;              //
Check Amount.
        } CIRS_CHECK;
```

```
// Complete set of fields recognized from an invoice
image.
typedef struct {
        CIRS_REC_RESULT          Amount;              //
Invoice Amount.
        CIRS_REC_RESULT          Paid;                //
Invoice Paid Amount.
        CIRS_REC_RESULT          Date;                //
Invoice Date.
        CIRS_REC_RESULT          Account;             //
Invoice Account.
} CIRS_INVOICE;


// All actions which can be returned from the Rules
DLL.
typedef CIRS_ACTION int;
typedef enum {
        CIRS_ACTION_ACCEPT,          // Good transaction.
        CIRS_ACTION_REJECT,          // Generic failed
                                     transaction.
        CIRS_ACTION_BAD_CHECK_IMAGE,    // Check image
is poor.
        CIRS_ACTION_BAD_CHECK_CAR,          // CAR
can't be
                                     validated.
        CIRS_ACTION_BAD_CHECK_LAR,          // LAR
can't be
                                     validated.
        CIRS_ACTION_BAD_CHECK_DATE,         // Check
Date can't be
                                     validated.
        CIRS_ACTION_BAD_CHECK_ACCT,         // Check
Acct can't be
                                     validated.
        CIRS_ACTION_BAD_CHECK_MICR,         // Check
MICR can't be
```

```
                                              validated.
        CIRS_ACTION_BAD_CHECK_FRAUD,          // Possibly
fraudulent

                                              check.
 5      CIRS_ACTION_BAD_INVOICE_IMAGE,        // Invoice
image is

                                              poor.
        CIRS_ACTION_BAD_INVOICE_AMOUNT,       // Invoice Amt
can't be
10                                            validated.
        CIRS_ACTION_BAD_INVOICE_PAID,         // Invoice Paid
can't be

                                              validated.
        CIRS_ACTION_BAD_INVOICE_DATE,         // Invoice Date
15 can't be

                                              validated.
        CIRS_ACTION_BAD_INVOICE_ACCT          // Invoice Acct
can't be

                                              validated.
20 } CIRS_ACTIONS;


// Confidence threshold used for validating fields.
Threshold values are determined experimentally,
// based on a large set of sample images.  Threshold
values are customized for each application.
25 typedef struct {
        CIRS_CONF       Thresh;          // 0-1000.
} CIRS_CONF_CONFIG;


// Set of thresholds for all fields on a check image.
typedef struct {
30      CIRS_CONF_CONFIG        CAR;
        CIRS_CONF_CONFIG        LAR;
        CIRS_CONF_CONFIG        Date;
        CIRS_CONF_CONFIG        Account;
CIRS_CONF_CONFIG        MagMICR;
```

```
              CIRS_CONF_CONFIG        OptMICR;
              CIRS_AMOUNT             MaxCashed;
        } CIRS_CHECK_CONFIG;


        // Set of thresholds for all fields on an invoice
  5     image.
        typedef struct {
              CIRS_CONF_CONFIG        Amount;
              CIRS_CONF_CONFIG        Paid;
              CIRS_CONF_CONFIG        Date;
 10           CIRS_CONF_CONFIG        Account;
        } CIRS_INVOICE_CONFIG;


        // Complete set of application and user-specific
        parameters.
        typedef struct {
 15     CIRS_CHECK_CONFIG       Check;
              CIRS_INVOICE_CONFIG Invoice;
        } CIRS_CONFIG;


        // Entry point for rules related to remittance
        processing.
 20     extern CIRS_ACTION CIRS_Rules_Remittance (
              CIRS_ENTERED    *pEntered;      // Fields entered by
        the user.
              CIRS_CHECK            *pCheck;         // Check image
        recognition.
 25           CIRS_INVOICE    *pInvoice;      // Invoice image
        recognition.
              CIRS_CONFIG           *pConfig);       // Application
        specific
                                              parameters.


 30     // Entry point for rules related to payroll check
        cashing..
        extern CIRS_ACTION CIRS_Rules_payroll (
```

```
        CIRS_ENTERED     *pEntered;        // Fields entered by
the user.
        CIRS_CHECK              *pCheck;          // Check image
recognition.
5       CIRS_CONFIG            *pConfig);        // Application
specific
                                        parameters.


// Entry point for rules related to proof of deposit..
extern CIRS_ACTION CIRS_Rules_POD (
10      CIRS_ENTERED     *pEntered;        // Fields entered by
the user.
        CIRS_CHECK              *pCheck;          // Check image
recognition.
        CIRS_CONFIG            *pConfig);        // Application
15 specific
                                        parameters.
```

        To facilitate the implementation of the Rules DLL,
an API is provided which supports a basic set of CIRS
related functions. The definition of the CIRS Support
20 API follows:

```
void CIRS_FilterByConf (
        CIRS_REC_RESULT     *pOutResult,
        CIRS_REC_RESULT     InResult,
        CIRS_CONF           Thresh);

25 void CIRS_SortByConf (
        CIRS_REC_RESULT     *pOutResult,
        CIRS_REC_RESULT     InResult);

int CIRS_ValidateMICR (
        CIRS_CANDIDATE      MICR,
30      CIRS_AMOUNT         Amount);
```

Remittance Processing

```
CIRS_ACTION CIRS_Rules_Remittance (
```

```
        CIRS_ENTERED        *pEntered;  // Fields entered by the user.
        CIRS_CHECK          *pCheck;    // Check image recognition.
        CIRS_INVOICE        *pInvoice;  // Invoice image recognition.
        CIRS_CONFIG         *pConfig)   // Application specific
5                                              parameters.


        {
        CIRS_RESULT                     CAR, LAR, CheckDate,
                                        CheckAcct, InvPaid, InvAcct;
        int                             i, found;


10  // Establish the list of candidates which pass threshold, for each
    field.
        CIRS_FilterByConf (&CAR, pCheck->CAR,
    pConfig->Check.CAR.Thresh);
        CIRS_FilterByConf (&LAR, pCheck->LAR,
15  pConfig->Check.LAR.Thresh);
        CIRS_FilterByConf (&CheckDate , pCheck->Date,
    pConfig->Check.Date.Thresh);
        CIRS_FilterByConf (&CheckAcct , pCheck->Account,
    pConfig->Check.Account.Thresh);
20      CIRS_FilterByConf (&InvPaid , pInvoice->Paid,
    pConfig->Invoice.Paid.Thresh);
        CIRS_FilterByConf (&InvAcct , pInvoice->Account,
    pConfig->Invoice.Account.Thresh);


    // Reject the transaction if there isnÆt at least one candidate
25  for each field.
        if (CAR.CandidateCount <= 0) return
    (CIRS_ACTION_BAD_CHECK_IMAGE);
        if (LAR.CandidateCount <= 0) return
    (CIRS_ACTION_BAD_CHECK_IMAGE);
30      if (CheckDate.CandidateCount <= 0) return
    (CIRS_ACTION_BAD_CHECK_IMAGE
        if (CheckAcct.CandidateCount <= 0) return
    (CIRS_ACTION_BAD_CHECK_IMAGE);
        if (InvPaid.CandidateCount <= 0) return (CIRS_ACTION_BAD_
35  INVOICE _IMAGE);
        if (InvAcct.CandidateCount <= 0) return (CIRS_ACTION_BAD_
    INVOICE _IMAGE);


    // At least one good date candidate must be within the last
    15-days.
```

```
          found = FALSE;
          for (i = 0; i < CheckDate.CandidateCount; i++) {
                 if ((CheckDate.Candidates(i).Value.Date.Date < time
     (NULL) &&
                         (CheckDate.Candidates(i).Value.Date.Date > (time
     (NULL) û 60*60*24*15))) {
                              found = TRUE;
                              break;
                 }
          }
          if (! found) return (CIRS_ACTION_BAD_CHECK_DATE);


     // At least one good candidate from the CAR field must match the
     Entered Paid Amount.
          found = FALSE;
          for (i = 0; i < CAR.CandidateCount; i++) {
                 if (CAR.Candidates(i).Value.Amount.Dollars ==
     pEntered->Paid.Dollars) {
                              found = TRUE;
                              break;
                 }
          }
          if (! found) return (CIRS_ACTION_BAD_CHECK_CAR);


     // At least one good candidate from the LAR field must match the
     Entered Paid Amount.
          found = FALSE;
          for (i = 0; i < LAR.CandidateCount; i++) {
                 if (LAR.Candidates(i).Value.Amount.Dollars ==
     pEntered->Paid.Dollars) {
                              found = TRUE;
                              break;
                 }
          }
          if (! found) return (CIRS_ACTION_BAD_CHECK_LAR);


     // At least one good candidate from the Invoice Paid field must
     match the Entered Paid Amount.
          found = FALSE;
          for (i = 0; i < InvPaid.CandidateCount; i++) {
                 if (InvPaid.Candidates(i).Value.Amount.Dollars ==
     pEntered->Paid.Dollars) {
                              found = TRUE;
```

5

10

15

20

25

30

35

40

```
                        break;
                    }
                }
            if (! found) return (CIRS_ACTION_BAD_INVOICE_PAID);

 5  // Transaction is acceptable.
            return (CIRS_ACTION_ACCEPT);
        }


    Payroll Check Cashing


    CIRS_ACTION CIRS_Rules_Payroll (
10          CIRS_ENTERED      *pEntered;  // Fields entered by the user.
            CIRS_CHECK        *pCheck;    // Check image recognition.
            CIRS_CONFIG       *pConfig)   // Application specific
                                             parameters.

    {
15          CIRS_RESULT                   MagMICR, OptMICR, CheckDate;
            int                           i, found;
            int                           Dollars, Cents;

    // Establish the list of candidates which pass threshold, for each
    field.
20          CIRS_FilterByConf (&MagMICR, pCheck->MagMICR,
        pConfig->Check.MagMICR.Thresh);
            CIRS_FilterByConf (&OptMICR, pCheck->OptMICR,
        pConfig->Check.OptMICR.Thresh);
            CIRS_FilterByConf (&CheckDate , pCheck->Date,
25      pConfig->Check.Date.Thresh);

    // Reject the transaction if there isnÆt at least one candidate
    for each field.
            if (MagMICR.CandidateCount <= 0) return
    (CIRS_ACTION_BAD_CHECK_IMAGE);
30          if (OptMICR.CandidateCount <= 0) return
    (CIRS_ACTION_BAD_CHECK_IMAGE);
            if (CheckDate.CandidateCount <= 0) return
    (CIRS_ACTION_BAD_CHECK_IMAGE

    // At least one good date candidate must be within the last 30
35  days.
            found = FALSE;
```

```
        for (i = 0; i < CheckDate.CandidateCount; i++) {
                if ((CheckDate.Candidates(i).Value.Date.Date < time
     (NULL) &&
                        (CheckDate.Candidates(i).Value.Date.Date > (time
5    (NULL) û 60*60*24*30))) {
                        found = TRUE;
                        break;
                }
        }
10      if (! found) return (CIRS_ACTION_BAD_CHECK_DATE);


     // At least one good candidate from the Optical MICR result must
     match the Magnetic MICR.
        Dollars = MagMICR.Candidates(0).Value.Amount.Dollars;
        Cents = MagMICR.Candidates(0).Value.Amount.Cents;
15      found = FALSE;
        for (i = 0; i < OptMICR.CandidateCount; i++) {
                if ((OptMICR.Candidates(i).Value.Amount.Dollars ==
     Dollars) &&
            (OptMICR.Candidates(i).Value.Amount.Cents == Cents)) {
20                      found = TRUE;
                        break;
                }
        }
        if (! found) return (CIRS_ACTION_BAD_CHECK_MICR);

25   // Check the MICR amount and account against the database.
        if (! CIRS_ValidateMICR (MagMICR.Candidates(0),
     pEntered->Cash.Dollars))
                return (CIRS_ACTION_BAD_CHECK_MICR);


     // Transaction is acceptable.
30      return (CIRS_ACTION_ACCEPT);
     }




     Proof Of Deposit


     CIRS_ACTION CIRS_Rules_POD (
            CIRS_ENTERED       *pEntered;  // Fields entered by the user.
35          CIRS_CHECK         *pCheck;    // Check image recognition.
            CIRS_CONFIG        *pConfig)   // Application specific
```

parameters.

```
{
        CIRS_RESULT      CAR, LAR, CheckDate;
        int              i, found;
5       int              BestCAR, BestLAR;


// Establish the list of candidates which pass threshold, for each
field.
        CIRS_FilterByConf (&CAR, pCheck->CAR,
pConfig->Check.CAR.Thresh);
10      CIRS_FilterByConf (&LAR, pCheck->LAR,
pConfig->Check.LAR.Thresh);
        CIRS_FilterByConf (&CheckDate , pCheck->Date,
pConfig->Check.Date.Thresh);


// Reject the transaction if there isnÆt at least one candidate
15 for each field.
        if (CAR.CandidateCount <= 0) return
(CIRS_ACTION_BAD_CHECK_IMAGE);
        if (LAR.CandidateCount <= 0) return
(CIRS_ACTION_BAD_CHECK_IMAGE);
20      if (CheckDate.CandidateCount <= 0) return
(CIRS_ACTION_BAD_CHECK_IMAGE


// At least one good date candidate must be within the last 30
days.
        found = FALSE;
25      for (i = 0; i < CheckDate.CandidateCount; i++) {
                if ((CheckDate.Candidates(i).Value.Date.Date < time
(NULL) &&
                        (CheckDate.Candidates(i).Value.Date.Date > (time
(NULL) û 60*60*24*30))) {
30                      found = TRUE;
                        break;
                }
        }
        if (! found) return (CIRS_ACTION_BAD_CHECK_DATE);


35 // At least one good candidate from the CAR field must match the
Entered Amount.
        found = FALSE;
        for (i = 0; i < CAR.CandidateCount; i++) {
```

```
                if (CAR.Candidates(i).Value.Amount.Dollars ==
        pEntered->Paid.Dollars) {
                        found = TRUE;
                        BestCAR = i;
 5                      break;
                }
            }
            if (! found) return (CIRS_ACTION_BAD_CHECK_CAR);


        // At least one good candidate from the LAR field must match the
10      Entered Amount.
            found = FALSE;
            for (i = 0; i < LAR.CandidateCount; i++) {
                if (LAR.Candidates(i).Value.Amount.Dollars ==
        pEntered->Paid.Dollars) {
15                      found = TRUE;
                        BestLAR = i;
                        break;
                }
            }
20          if (! found) return (CIRS_ACTION_BAD_CHECK_LAR);


        // Detect possible check tampering.
            if ((CAR.Candidates(BestCAR).Conf > CarFraudMin) &&
                (LAR.Candidates(BestLAR).Conf -
        LAR.Candidates(BestLAR).Words(0).Conf >
25          LarFraudSpread))
            return (CIRS_ACTION_BAD_CHECK_FRAUD);


        // Transaction is acceptable.
            return (CIRS_ACTION_ACCEPT);



        Proof Of Deposit


30      CIRS_ACTION CIRS_Rules_Payroll (
            CIRS_ENTERED        *pEntered;  // Fields entered by the user.
            CIRS_CHECK          *pCheck;    // Check image recognition.
            CIRS_CONFIG         *pConfig)   // Application specific
                                            //   parameters.


35      {
            CIRS_RESULT     MagMICR, OptMICR, CheckDate;
```

```
int               i, found;
int               Dollars, Cents;


// Establish the list of candidates which pass threshold, for each
field.
        CIRS_FilterByConf (&MagMICR, pCheck->MagMICR,
pConfig->Check.MagMICR.Thresh);
        CIRS_FilterByConf (&OptMICR, pCheck->OptMICR,
pConfig->Check.OptMICR.Thresh);
        CIRS_FilterByConf (&CheckDate , pCheck->Date,
pConfig->Check.Date.Thresh);


// Reject the transaction if there isn't at least one candidate
for each field.
        if (MagMICR.CandidateCount <= 0) return
(CIRS_ACTION_BAD_CHECK_IMAGE);
        if (OptMICR.CandidateCount <= 0) return
(CIRS_ACTION_BAD_CHECK_IMAGE);
        if (CheckDate.CandidateCount <= 0) return
(CIRS_ACTION_BAD_CHECK_IMAGE


// At least one good date candidate must be within the last 30
days.
        found = FALSE;
        for (i = 0; i < CheckDate.CandidateCount; i++) {
                if ((CheckDate.Candidates(i).Value.Date.Date < time
(NULL) &&
                        (CheckDate.Candidates(i).Value.Date.Date > (time
(NULL) - 60*60*24*30))) {
                        found = TRUE;
                        break;
                }
        }
        if (! found) return (CIRS_ACTION_BAD_CHECK_DATE);


// At least one good candidate from the Optical MICR result must
match the Magnetic MICR.
        Dollars = MagMICR.Candidates(0).Value.Amount.Dollars;
        Cents = MagMICR.Candidates(0).Value.Amount.Cents;
        found = FALSE;
        for (i = 0; i < OptMICR.CandidateCount; i++) {
                if ((OptMICR.Candidates(i).Value.Amount.Dollars ==
Dollars) &&
```

```
(OptMICR.Candidates(i).Value.Amount.Cents == Cents)) {
                found = TRUE;
                break;
        }
    }
    if (! found) return (CIRS_ACTION_BAD_CHECK_MICR);

// Check the MICR amount and account against the database.
    if (! CIRS_ValidateMICR (MagMICR.Candidates(0),
pEntered->Cash.Dollars))
            return (CIRS_ACTION_BAD_CHECK_MICR);

// Transaction is acceptable.
        return (CIRS_ACTION_ACCEPT);
```

It will be appreciated that although various aspects of the invention have been described with respect to specific embodiments, alternatives and modifications will be apparent from the present disclosure, which are within the spirit and scope of the present invention as set forth in the following claims.